

THESIS / THÈSE

MASTER EN SCIENCES INFORMATIQUES

Surveillance épidémiologique par programmes informatisés à la clinique Saint-Jean de Bruxelles

Lefebvre, Pascale

Award date:
1987

Awarding institution:
Université de Namur

[Link to publication](#)

General rights

Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain
- You may freely distribute the URL identifying the publication in the public portal ?

Take down policy

If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

FACULTES
UNIVERSITAIRES
N.D. DE LA PAIX

NAMUR



INSTITUT D'INFORMATIQUE

SURVEILLANCE EPIDEMIOLOGIQUE PAR
PROGRAMMES INFORMATISES A LA CLINIQUE SAINT-JEAN
DE BRUXELLES

Pascale LEFEBVRE

Memoire presente en vue de
l'obtention du titre de
Licencie et Maitre en
informatique.

Promoteur : J. FICHEFET

Annee academique : 1986-1987

Je remercie toutes les personnes qui m'ont aidée à la réalisation de ce travail.

Au Professeur J. Fichet qui m'a permis de réaliser ce travail dans le domaine médical que j'apprécie.

Au Docteur Saille, chef de service de la clinique Saint-Jean, pour ses conseils et son vif intérêt au projet.

Je remercie également les membres du groupe "labo" dont je fais partie et plus particulièrement le chef de projet "labo", Monsieur Christian Pauly, pour toute l'aide qu'ils m'ont apportée.

PRESENTATION

PARTIE 1 : INTRODUCTION

PARTIE 2 : ETUDE ET CONTEXTE DU PROJET

I. L'EPIDEMIOLOGIE

1. DEFINITIONS
2. PRINCIPES DE BASE
3. METHODES EPIDEMIOLOGIQUES

II. LE PROJET DE SURVEILLANCE EPIDEMIOLOGIQUE A SAINT-JEAN

1. METHODE UTILISEE
2. LIMITES DU PROJET

PARTIE 3 : ANALYSE FONCTIONNELLE ET ORGANIQUE

I. DEMARCHE D'ANALYSE

II. ETUDE D'OPPORTUNITE

1. BESOINS SOUHAITES PAR LE COMITE D'HYGIENE
2. BESOINS SOUHAITES PAR LE MEDECIN DU SERVICE DE REANIMATION
3. BESOINS SOUHAITES PAR LE BACTERIOLOGISTE EPIDEMIOLOGISTE

III. ETUDE DE L'EXISTANT

1. ETUDE ET CONTEXTE DU PROJET AU NIVEAU DU LABORATOIRE
2. ETUDE ET CONTEXTE DU PROJET EN BACTERIOLOGIE
3. ETUDE ET CONTEXTE DU PROJET AU NIVEAU DE L'ADMISSION

IV. ETUDE ET DESCRIPTION DE LA SOLUTION PROJETEE

0. INTRODUCTION
1. APPLICATIONS BACTERIOLOGIE
2. APPLICATION EPIDEMIOLOGIE

V. ANALYSE FONCTIONNELLE

1. DESCRIPTION DES DONNEES
2. DESCRIPTION DES TRAITEMENTS

VI. ANALYSE ORGANIQUE

1. PRESENTATION DES OUTILS UTILISES
2. SCHEMA DES ACCES POSSIBLES
3. ARCHITECTURE DES TRAITEMENTS
4. SCHEMA DES ACCES NECESSAIRES
5. CHOIX DES LANGAGES

PARTIE 4 : CONCLUSIONS ET PERSPECTIVES

PARTIE 5 : BIBLIOGRAPHIE

PARTIE 6 : ANNEXES

TABLE DES MATIERES

PARTIE 1 : INTRODUCTION	1
PARTIE 2 : ETUDE ET CONTEXTE DU PROJET	4
I. L'EPIDEMIOLOGIE	4
1. DEFINITIONS	4
1.1. Epidémiologie	4
1.2. Infections	4
2. PRINCIPES DE BASE	5
2.1. L'agent	5
2.2. La transmission	5
2.2.1. Contact Spread	5
2.2.2. Common Vehicle Spread	6
2.2.3. Air Born Spread	6
2.2.4. Vector Born Spread	6
2.3. L'hôte	6
2.3.1. Site de multiplication de l'agent infectieux	7
2.3.2. Mécanismes de défense spécifiques	7
2.3.3. Mécanismes de défense non spécifiques	7
2.4. L'environnement	7
3. METHODES EPIDEMIOLOGIQUES	8
3.1. Etude épidémiologique descriptive	8
3.1.1. Temps	8
3.1.2. Place	9
3.1.3. Personne	9
3.2. Etude épidémiologique analytique	9
3.2.1. CASE CONTROL STUDY	9
3.2.2. COHORT STUDY	10
3.3. Etude épidémiologique expérimentale	10
II. LE PROJET DE SURVEILLANCE EPIDEMIOLOGIQUE A SAINT-JEAN	11
1. METHODE UTILISEE	11
2. LIMITES DU PROJET	12

PARTIE 3 : ANALYSE FONCTIONNELLE ET ORGANIQUE	13
I. DEMARCHE D'ANALYSE	13
II. ETUDE D'OPPORTUNITE	15
1. BESOINS SOUHAITES PAR LA COMMISSION D'HYGIENE	15
2. BESOINS SOUHAITES PAR LE MEDECIN DU SERVICE DE REANIMATION	15
3. BESOINS SOUHAITES PAR LE BACTERIOLOGISTE EPIDEMIOLOGISTE	16
III. ETUDE DE L'EXISTANT	16
1. ETUDE ET CONTEXTE DU PROJET AU NIVEAU DU LABORATOIRE	16
1.1. Encodage de la demande et prélèvement des échantillons	17
1.1.1. Demandes "consultants" et "apportées"	17
1.1.2. Demandes "hospitalisés"	18
1.1.2.1. Demande d'analyse de sang non urgente	18
1.1.2.2. Demande d'analyse sur liquides divers non urgente	18
1.1.2.3. Demande d'analyse urgente	18
1.2. Réception des échantillons au dispatch du laboratoire	18
1.3. Dosage des analyses par station de travail	19
1.4. Validation des résultats et édition des protocoles	19
2. ETUDE ET CONTEXTE DU PROJET EN BACTERIOLOGIE	19
2.0. Introduction	19
2.1. Encodage de la demande et prélèvement des échantillons	20
2.2. Réception des échantillons au dispatch du laboratoire	21
2.3. Dosage des analyses par station de travail	21
2.4. Validation médicale des résultats et édition des protocoles	22
3. ETUDE ET CONTEXTE DU PROJET AU NIVEAU DE L'ADMISSION	22
3.1. Notions d'admission et de séjour	22
3.2. Suivi d'une demande d'hospitalisation	22
3.2.1. Admission	22
3.2.1.1. Admission non urgente	23
3.2.1.2. Admission urgente	23
3.2.2. Autres transferts	24
3.2.2.1. Transfert de lit	24
3.2.2.2. Prolongation de séjour	24
3.2.2.3. Sortie du patient	24

IV. ETUDE ET DESCRIPTION DE LA SOLUTION PROJETEE	24
0. INTRODUCTION	24
1. APPLICATION BACTERIOLOGIE	26
1.1. Nouvelles structures des résultats bactériologie et leurs implications	26
1.1.1. Encodage de la demande et prélèvement des échantillons	27
1.1.2. Réception des échantillons au dispatch du laboratoire	28
1.1.3. Dosage des analyses par station de travail	28
1.1.4. Validation médicale des résultats et édition des protocoles	28
2. APPLICATION EPIDEMIOLOGIE	29
2.1. Objectifs à réaliser	29
2.2. Définitions et concepts	29
2.2.0. Notions body-sites "master/slave"	29
2.2.1. Pathogénicité d'une infection	30
2.2.2. Cross-infection	30
2.2.3. Délai de non redondance	30
2.2.4. Résultat non redondant	30
2.2.5. Résultat bactériologique à traiter	30
2.3. Traitement journalier	31
2.3.1. Test de redondance	31
2.3.2. Test de pathogénicité	31
2.3.3. Test si cible	32
2.3.4. Format des listes journalières	33
2.4. Traitement "a posteriori"	33
2.4.1. Liste des patients infectés triée par germe et par service	33
2.4.2. Liste des patients infectés triée par service et par germe	34
2.4.3. Liste de répartition des germes par service	35
2.4.4. Taux d'infection par service	35
2.4.5. Taux d'infection par body-service et par service	36
2.4.6. Histogramme de sensibilité d'un germe	36
2.4.7. Polyrésistance des germes aux antibiotiques	37
V. ANALYSE FONCTIONNELLE	39
1. DESCRIPTION DES DONNEES	39
1.1. Méthodologie	39
1.2. Modèle Entité-Association	40
1.3. Liste des propriétés par Entités et Associations	41
1.4. Contraintes d'intégrité	76
1.4.1. Contraintes d'existence	76
1.4.2. Contraintes d'inclusion	76

2. DESCRIPTION DES TRAITEMENTS	77
2.1. Méthodologie	77
2.2. Décomposition du projet en applications	77
2.3. Décomposition des applications en phases	77
2.3.1. Description et spécifications des phases	77
2.3.2. Les différentes phases	78
2.3.3. Description des phases spécifiées	80
2.4. Décomposition des phases en fonctions	99
2.4.1. Description et spécifications des fonctions	99
2.4.2. Description des fonctions	100
VI. ANALYSE ORGANIQUE	131
1. PRESENTATION DES OUTILS UTILISES	131
1.1. Hardware	131
1.1.1. Equipement	131
1.1.2. Fiches techniques	131
1.1.3. Configuration	131
1.2. Software	132
1.2.1. Système d'exploitation	132
1.2.2. Organisation des données	132
1.2.2.1. Thesauri	132
1.2.2.2. TVR	134
1.2.2.3. Random files	135
1.2.2.4. Autre fichiers	135
1.2.2.5. Taille des programmes	135
2. SCHEMA DES ACCES POSSIBLES	136
2.1. Méthodologie	136
2.2. Schéma des accès possibles	136
3. ARCHITECTURE DES TRAITEMENTS	136
3.1. Méthodologie	136
3.2. Architecture des traitements	137
3.3. Description des différents niveaux	138
4. SCHEMA DES ACCES NECESSAIRES	138
4.1. Méthodologie	138
4.2. Organisation des fichiers implémentés	138
4.3. Schéma des accès nécessaires	139
5. CHOIX DES LANGAGES	139
PARTIE 4 : CONCLUSIONS ET PERSPECTIVES	140
PARTIE 5 : BIBLIOGRAPHIE	142
PARTIE 6 : ANNEXES	

PARTIE 1 : INTRODUCTION

Depuis deux ans, je travaille au service informatique de la clinique Saint-Jean de Bruxelles, dans le cadre du projet laboratoire.

Comme travail de fin d'études, j'ai choisi la réalisation de programmes informatisés de surveillance des infections hospitalières.

Les raisons essentielles de ce choix sont :

- l'importance d'une telle surveillance dans un hôpital moderne
- l'énorme travail que celle-ci nécessiterait s'il fallait la réaliser manuellement.

A la jeune biologiste et informaticienne que je suis, ce travail m'a donné l'occasion de découvrir le cycle de vie complet du développement d'un projet, qui non seulement se situe dans un environnement connu, mais qui en plus allie mes connaissances en biologie médicale à l'intérêt que je porte à l'informatique.

Les laboratoires de microbiologie ont une responsabilité importante dans la surveillance, le contrôle et la prévention des infections hospitalières. Un des principaux rôles des bactériologistes et de la commission d'hygiène est de définir avec l'aide des cliniciens :

- 1) les infections ou les colonisations
- 2) le caractère hospitalier de ces infections

Ce double but réalisé, on peut d'une part :

- prendre des mesures le plus rapidement possible dès qu'un patient est infecté
- connaître le plus rapidement possible la propagation des infections hospitalières

et d'autre part :

- mieux évaluer l'hygiène du personnel ou du matériel utilisé
- mieux connaître la qualité des soins
- étudier plus facilement la cause de la propagation des infections hospitalières

L'intérêt que suscite cette étude n'est pas sans conséquences au niveau même du coût du séjour hospitalier étant donné la prolongation du séjour et les soins supplémentaires à prodiguer au patient infecté. On estime par exemple, que dans un cas sur six, la prolongation atteint 2 à 3 jours pour un hôpital moderne, voire 8 à 15 jours pour 100 malades. Tout laisse à penser aussi que dans un avenir plus ou moins proche, des bandes magnétiques devront être envoyées au Ministère de la Santé qui permettront de

comparer entre eux les hôpitaux au niveau de leur hygiène hospitalière.

Pour parvenir à ces buts, l'efficacité des systèmes informatiques est indéniable. Elle permet non seulement de mémoriser un grand nombre de données qui peuvent être sélectionnées, triées, éditées en un minimum de temps, mais elle a aussi l'avantage de fournir rapidement aux cliniciens des statistiques, des informations médicales concernant des patients infectés. L'informatique peut aussi fournir chaque jour l'instantané de la situation infectieuse d'un service, c'est-à-dire, l'analyse en temps réel de celle-ci. De cette façon, certaines mesures d'hygiène ne seront plus appliquées en aveugle ou avec retard et verront croître leur efficacité.

La réalisation de ce travail nous a permis de découvrir toute la difficulté d'intégrer une nouvelle application dans un environnement réel, avec ses contraintes, ses exigences, ses limites et ses structures établies.

Notre démarche a commencé par une étude plus théorique de l'épidémiologie, de ses concepts et de ses méthodes. Après nous être familiarisés avec les notions, nous avons décrit le contexte réel dans lequel le projet sera intégré.

Cette description commence par l'étude du suivi d'une demande depuis sa prescription par le médecin jusqu'à l'édition des résultats des analyses sur protocoles, en passant par l'encodage de la demande et la validation par le médecin.

De la même manière, nous nous sommes intéressés aux demandes d'analyses bactériologiques étant donné l'importance jouée par celles-ci dans le cadre de l'épidémiologie.

Parallèlement à cette étude, nous avons décrit le contexte plus hospitalier de la gestion des lits, depuis l'admission du patient jusqu'à son départ de la clinique.

Dès lors, quatre grandes parties se dessinent dans cette étude :

La première étape consiste à étudier le contexte du projet au niveau du laboratoire en général, du laboratoire de bactériologie plus spécifiquement et de la gestion des lits.

La seconde étape spécifie clairement les traitements à réaliser et les données à exploiter.

La troisième étape définit les règles de traitements et la structure des données.

La quatrième étape est la réalisation, l'implémentation de ces traitements.

Pour terminer, en guise de conclusion, nous tenterons de définir l'avenir de ce projet, ses applications futures et nous proposerons d'autres applications susceptibles d'apporter d'autres renseignements pour apprécier et mieux comprendre l'évolution du taux d'infection dans l'hôpital.

PARTIE 2 : ETUDE ET CONTEXTE DU PROJET

I. L'EPIDEMIOLOGIE

1. DEFINITIONS

1.1. Epidémiologie

L'épidémiologie est la science qui étudie les lois objectives de l'apparition, de la propagation et de la fin des maladies transmissibles dans une collectivité humaine et qui met au point les mesures de prophylaxie et de lutte contre ces maladies.

Au symposium d'épidémiologie international de Prague (1960) la définition suivante fut adoptée à l'unanimité : "L'épidémiologie est une branche autonome de la médecine ayant pour but la recherche des causes pour lesquelles surviennent et se propagent les maladies infectieuses dans la société humaine et appliquant les connaissances acquises pour combattre, prévenir et finalement, éradiquer totalement ces maladies."

A cette définition, s'ajoute la notion d'infection.

1.2. Infections

Le comité restreint d'experts du conseil de l'Europe a, dans l'exposé des motifs de la réunion, tenue à Strasbourg du 18 au 26 octobre 1984, défini de façon suivante l'infection hospitalière dite infection nosocomiale : "L'infection contractée à l'hôpital se réfère à toute maladie due à des micro-organismes, cliniquement et/ou microbiologiquement reconnaissables, qui affecte le malade du fait de son admission à l'hôpital ou des soins qu'il a reçus en tant que patient hospitalisé ou en traitement ambulatoire, soit le personnel hospitalier du fait de son activité, que les symptômes apparaissent ou non pendant le séjour de l'intéressé".

Le mot infection est pris dans le sens de la multiplication de micro-organismes avec manifestations localisées et/ou générales et doit être opposé à :

- Contamination : Processus entraînant la présence de micro-organismes sur le matériel ou la personne.
- Inoculation : Introduction de micro-organismes susceptibles de se multiplier dans les tissus.

- Colonisation : Multiplication localisée de germes qui peut résulter d'une contamination ou d'une inoculation, sans réaction tissulaire et qui devient partie de la flore du sujet.

2. PRINCIPES DE BASE

Pour mieux comprendre le processus de l'infection, étudions de manière plus systématique la chaîne de l'infection qui est constituée de trois éléments :

- l'agent
- la transmission
- l'hôte

En connaissant les caractéristiques de l'agent, les différents modes de transmission possibles, les caractéristiques de l'hôte chez qui l'infection peut se manifester, et l'environnement dans lequel cette chaîne évoluerait, nous pourrions connaître les données nécessaires et suffisantes pour aider l'épidémiologiste à détecter et analyser l'infection.

2.1. L'agent

L'agent infectieux peut être une bactérie, un virus, un parasite, une levure, mais la plupart des infections hospitalières sont dues à un virus ou une bactérie. Les caractéristiques de l'agent sont :

- la pathogénicité qui mesure la capacité d'induire une maladie
- la virulence qui mesure la sévérité de la maladie
- l'invasion qui décrit la capacité de l'agent à envahir les tissus
- la dose infectieuse qui est la quantité nécessaire pour causer l'infection
- la spécificité
- la période d'incubation
- les caractéristiques biochimiques et génétiques

2.2. La transmission

De manière grossière on distingue quatre grands modes de transmission :

- "CONTACT SPREAD"
- "COMMON VEHICLE SPREAD"
- "AIR BORN SPREAD"
- "VECTOR BORN SPREAD"

2.2.1. CONTACT SPREAD

Dans ce premier mode, la victime est en contact avec la source. Ce contact peut être direct ou indirect. Lors du contact direct, il y a relation intrinsèque entre l'agent et la victime. (ex : origine féco-orale de l'hépatite B).

Il y a contact indirect, quand la situation est telle que

la victime rentre en contact avec l'objet intermédiaire contaminé qui induit l'infection.

Ex : matériel endoscopique contaminé
contamination de la maladie par la toux ou la parole
dans le cas où l'objet intermédiaire inanimé est
l'air ambiant

2.2.2. COMMON VEHICLE SPREAD

Dans ce mode de transmission, le véhicule inanimé contaminé sert de vecteur de transmission de l'agent.

Ex : infection des salmonella qui se multiplient dans la nourriture

2.2.3. AIR BORN SPREAD

La transmission via l'air ambiant décrit les organismes qui ont une phase de dissémination dans l'air. Généralement les distances entre l'origine et la victime sont supérieures à quelques mètres. Les organismes sont contenus dans des gouttelettes ou sont fixés à des poussières pendant des périodes de temps parfois prolongées et transmettent ainsi l'agent.

Ex : infection transmise par courant d'air dans des couloirs, par l'air confiné dans certaines pièces surchauffées...

2.2.4. VECTOR BORN SPREAD

Ce quatrième mode de transmission, inclut les vecteurs de transmission internes et externes.

Vecteurs de transmission externe : Le transfert mécanique de micro-organisme sur l'hôte (ex : la mouche transporte l'agent infectieux)

Vecteurs de transmission interne : Le transfert inclut la transmission biologique et "l'haborage".

Dans la transmission biologique l'agent subit des changements au sein même du vecteur (ex : le plasmodium subit des modifications dans les glandes salivaires de l'anophèle avant qu'elle ne pique et infecte l'hôte).

Par contre dans "l'haborage" aucune action biologique ne se manifeste entre le vecteur et l'agent.

2.3. L'hôte

La maladie ne se développe pas systématiquement quand l'agent infectieux a été transmis à l'hôte. En effet, une variété de facteurs de l'hôte jouent un rôle et influencent le développement ou non de la maladie. Ces facteurs sont :

- le site de multiplication de l'agent infectieux
- les mécanismes de défense spécifiques
- les mécanismes de défense non spécifiques

2.3.1. Site de multiplication de l'agent infectieux

Du site de multiplication de l'agent infectieux dépend la pathogénicité de la maladie.

Ex : il est normal de trouver des E.Coli dans les selles, mais anormal et pathogène de les trouver dans les voies urinaires à partir d'une certaine quantité

2.3.2. Mécanismes de défense spécifiques

Parmi les mécanismes de défense spécifiques citons :

- l'immunité naturelle qui dépend des maladies antérieures contractées
- l'immunité artificielle active ou passive acquise par l'administration de vaccins ou de sérum avec anticorps

2.3.3. Mécanismes de défense non spécifiques

Les mécanismes de défense non spécifiques incluent la peau, les muqueuses, les sécrétions, qui sont des barrières mécaniques ou ont une action antimicrobienne contre les agents pathogènes.

D'autres facteurs non spécifiques tels que l'âge, les maladies contractées précédemment, les opérations subies, ont également une influence sur le développement de la maladie contractée.

2.4. L'environnement

Toute la chaîne de l'infection évolue dans un environnement qui peut aussi jouer son rôle à plusieurs niveaux. En effet, toute une série de facteurs s'influencent, et un changement de l'un d'eux peut perturber la chaîne. Certains facteurs de l'environnement peuvent être contrôlés comme l'air ambiant ou les surfaces des salles d'opérations. D'autres comme le degré d'humidité de l'air, la circulation, ne peuvent l'être.

Citons quelques exemples éloquentes où l'environnement influence la vie de la chaîne ou de l'un de ses chaînons :

- l'humidité de l'air, ou la sécheresse, affecte la multiplication de l'agent infectieux, en influençant sa persistance à la source, sa transmission dans l'air plus facile et le manque d'efficacité des muqueuses de l'hôte.

- la vitesse, ou la direction de l'air influence la transmission "AIR BORN"

Ex : en hiver, l'atmosphère confinée, la circulation d'air minimale, favorise la suspension des microbes.

- les conditions climatiques influencent le mouvement, la multiplication du vecteur de l'agent pathogène

Ex : conditions climatiques humides et chaudes en Afrique Australe favorisent la multiplication de l'anophèle, vecteur du plasmodium (agent pathogène responsable de la malaria).

3. METHODES EPIDEMIOLOGIQUES

On distingue trois méthodes pour étudier l'épidémiologie :

- Etude épidémiologique descriptive
- Etude épidémiologique analytique
- Etude épidémiologique expérimentale

3.1. Etude épidémiologique descriptive

Dans la plupart des cas, l'étude descriptive est la première étude réalisée dans le cadre d'une investigation épidémiologique. L'épidémiologie descriptive, décrit les occurrences des maladies en termes de temps, de place et de personnes. Le problème une fois ainsi défini, les deux autres méthodes permettent d'obtenir des données supplémentaires afin de confirmer ou d'infirmer les impressions initiales ou les hypothèses avancées.

3.1.1. Temps

On peut étudier comment varie l'évolution de la maladie en fonction du temps. On constate alors qu'il y a trois tendances :

- Evolution séculaire : Elle s'évalue sur une période de plusieurs années (>20 ans). Elle reflète les niveaux socio-économiques ou nutritionnels ou immunitaires d'une population.

- Evolution périodique : Elle s'évalue sur quelques années et peut être considérée comme des interruptions de l'évolution séculaire.

- Evolution saisonnière : Comme son nom le laisse supposer, les occurrences des cas fluctuent dans l'année. Celles-ci sont très souvent en rapport avec les circonstances qui influencent le mode de transmission de l'agent.

- Evolution spontanée : L'évolution est tout à fait imprévisible, et l'occurrence des cas est sporadique. Dans ces cas, l'évolution de la maladie dépend de différents facteurs tels que :

- les caractéristiques de l'agent
- la pathogénicité de l'agent
- la concentration de l'agent
- la période d'incubation
- le mode de transmission

- les caractéristiques de l'hôte
- les facteurs de l'environnement

3.1.2. Place

L'étude descriptive concerne aussi l'aire géographique où l'agent et l'hôte ont été en contact. Dans l'hôpital on parlera de service, de chambre, de lit ou de salle. Remarquons que l'endroit où se trouve le patient actuellement n'est pas nécessairement le même que celui où l'infection a été contractée. Dès lors un suivi des patients dans les services ou les chambres doit pouvoir être connu.

3.1.3. Personne

L'étude descriptive se penche aussi sur les facteurs de l'hôte tels que : l'âge, le sexe, les maladies contractées antérieurement, l'immunité, la thérapeutique subie, la médication... En d'autres termes tous les facteurs qui peuvent influencer le développement de la maladie sont à prendre en considération. (ex : les bébés ou les personnes âgées sont plus fragiles et par conséquent plus aptes à attraper l'infection).

3.2. Etude épidémiologique analytique

La méthode analytique étudie et évalue les relations entre les causes et les effets de la maladie. Cette méthode comprend deux méthodes de base :

- une méthode rétrospective (CASE CONTROL STUDY) qui part des effets pour arriver aux causes;
- une méthode prospective (COHORT STUDY) qui part des causes pour évaluer les effets.

La collecte des données pour ces deux méthodes est fondamentalement différente. Dans la première méthode, les données sont collectées après l'occurrence de la maladie, dans la seconde méthode, elles sont collectées au moment de l'occurrence de la maladie.

3.2.1. CASE CONTROL STUDY

Cette méthode analytique est appliquée sur un certain nombre de personnes subdivisée en deux groupes :

- groupe d'étude
- groupe contrôle de comparaison

Le premier groupe est l'ensemble des personnes affectées par la maladie X. Le second groupe est l'ensemble des personnes qui ne sont pas affectées par la maladie X. En

étudiant plus précisément les circonstances dans lesquelles se sont trouvées les personnes affectées, et par comparaison au second groupe, on peut mettre en évidence les causes possibles de la maladie X.

Cette méthode a pour inconvénient, la difficulté de récolter les données suffisantes pour répartir les patients en deux groupes significatifs. Elle est surtout utilisée lors d'un pic d'occurrence d'une maladie car le passage du médecin hygiéniste se fera souvent après la disparition de ce pic.

3.2.2. COHORT STUDY

Dans cette seconde méthode, les patients sont exposés à des "causes" et sont comparés au groupe de personnes non exposées aux "causes". Cette méthode nécessite une collecte des données et une étude poussée de celles-ci. Toutefois, elle fournit une estimation directe du risque d'infection de certains agents.

Inconvénients : Cette méthode est difficile à appliquer pour des maladies dont la période d'incubation est longue.

3.3. Etude épidémiologique expérimentale

Cette méthode est utilisée pour vérifier la justesse des hypothèses établies lors de l'étude d'une nouvelle méthode de prévention épidémiologique.

Soit l'hypothèse H, selon laquelle :

Le médicament X = prévention maladie Y

Soit deux groupes qui se distinguent uniquement par l'administration ou non du médicament X. L'étude se poursuit par l'utilisation de la méthode b1) ou b2) afin d'étudier l'interaction entre les causes et les effets. Ainsi, l'hypothèse H sera confirmée ou infirmée.

II. LE PROJET DE SURVEILLANCE EPIDEMIOLOGIQUE A SAINT-JEAN

1. METHODE UTILISEE

Parmi les méthodes épidémiologiques précédemment décrites, dans le cadre de la surveillance des infections hospitalières à Saint-Jean, la méthode descriptive sera retenue. En effet, cette méthode permet une étude satisfaisante du problème à partir des données disponibles tant au niveau du laboratoire qu'au niveau hospitalier ou administratif. Ceci est d'autant plus vrai que la volonté de la clinique Saint-Jean est de posséder un système totalement intégré. De plus, elle n'est pas restrictive et sans demander trop de temps d'investigation ni de trop gros changements dans la manière de travailler de la clinique, elle permet de détecter assez rapidement les cas possibles d'infections nosocomiales, et à partir d'eux, des mesures peuvent être prises dans le service.

En étudiant au cours du temps les occurrences des infections on peut évaluer l'évolution du taux d'infection de la clinique, ce qui est un marqueur de la qualité des soins administrés et de l'hygiène de la clinique en général.

En étudiant l'aire géographique où ont été détectées les infections, on pourra :

- comparer les infections enregistrées par services;
- localiser les infections et chercher plus facilement les causes possibles. Ex : matériel contaminé, personnel porteur d'agent infecté;
- suivre un patient infecté et voir l'influence que celui-ci exerce auprès des autres patients.

En étudiant le patient hôte, on peut mettre en évidence :

- l'influence de la thérapeutique utilisée;
- l'influence du sexe, de l'âge du patient sur le taux d'infection;
- l'influence des médicaments ou des antibiotiques prescrits pour ce patient;
- l'influence des "portes d'entrée" (catheter, i.V.,...) sur l'occurrence des infections.

Parallèlement à ces descriptions, on peut élaborer des statistiques utiles aux cliniciens telles que :

- statistiques du nombre de germes par site de prélèvement;
- nombre de germes rencontrés par service;
- pourcentage de résistance de certains germes par rapport à certains antibiotiques;
- ...

Des éditions journalières peuvent être très utiles telles que :

- les listes des candidats d'infections pathologiques par service;

- les listes des candidats d'infections croisées (un patient contamine un autre patient).
Notons que les infections croisées sont des cas particuliers d'infections nosocomiales.

2. LIMITES DU PROJET

Pour parvenir aux buts fixés, il est important de confronter les données du laboratoire avec les renseignements cliniques et administratifs. En d'autres termes, l'enregistrement des renseignements médicaux tels que les opérations ou thérapeutiques administrées, nous rend tributaires de l'encodage de ces données médicales. Toutefois, il faut signaler que l'encodage n'est réellement complet que si les données encodées ont une implication au niveau de la facturation. De plus, celui-ci est souvent réalisé après la sortie du patient de la clinique. Dès lors, dans le cadre de ce projet d'épidémiologie, nous nous limiterons à la confrontation des résultats labo et des renseignements administratifs qui concernent la gestion des lits c'est-à-dire l'admission.

Dans la conclusion de ce travail, nous expliciterons les développements futurs de ce projet qui dépendront d'une meilleure disponibilité des renseignements médicaux et d'une meilleure intégration de ceux-ci.

Dans la poursuite des buts décrits, nous sommes aussi tributaires des demandes d'analyse prescrites par le clinicien. Ceci est surtout vrai pour les catheters et écouvillonnages de plaies qui ne sont pas systématiquement envoyées au laboratoire en cas d'infection locales.

L'étude épidémiologique n'est possible que si des demandes de bactériologie sont prescrites dans les 24 heures après l'admission. Dans le cas contraire, il serait impossible de distinguer une infection nosocomiale, d'une infection déjà contractée à l'extérieur de la clinique.

On exclut de la surveillance des infections nosocomiales les infections virales. Dès lors, on pourrait se poser la question : "Pourquoi ne pas s'intéresser aux virus en tant qu'agent responsable des infections hospitalières ?"

Contrairement aux bactéries, les virus sont des micro-organismes dont la structure moléculaire évolue au cours du développement. Dès lors, ni des statistiques, ni des sélections, ni une étude de la propagation d'un certain virus ne sont à envisager sinon sur quelques virus stables tels que les virus de l'hépatite. Par ailleurs, une étude sur les virus exige une grande coopération de la part des spécialistes tels que les gastro-entérologues, les dermatologues, etc.

PARTIE 3 : ANALYSE FONCTIONNELLE ET ORGANIQUE

I. DEMARCHE D'ANALYSE

Afin de mieux comprendre le raisonnement suivi au cours de l'analyse du projet d'épidémiologie, précisons les sept grandes étapes suivantes :

- a) Etude d'opportunité
- b) Etude et critique de l'existant
- c) Etude et description de la solution projetée pour le projet épidémiologie
- d) Analyse fonctionnelle
- e) Analyse organique
- f) Implémentation
- g) Conclusions et perspectives

a) Etude d'opportunité

L'objectif de celle-ci est d'exprimer le plus clairement possible les raisons qui ont fait l'objet de cette étude.

b) Etude et critique de l'existant

La raison principale de cette étape est de comprendre le contexte dans lequel le projet doit être intégré. Le projet épidémiologie et les applications qui y seront réalisées, évoluent dans un système que nous décrirons en trois volets. Chaque volet sera décrit et supporté par un diagramme de flux ou un schéma de la dynamique des traitements. Celle-ci permettra de décrire clairement l'enchaînement des traitements.

1er volet : Suivi d'une demande d'analyse depuis sa prescription jusqu'à l'édition des résultats sur protocoles.

2e volet : Suivi d'une demande d'analyse de type bactériologie, depuis sa prescription jusqu'à l'édition des protocoles. Les résultats obtenus pour ces analyses nous intéressent plus particulièrement dans le cadre de cette étude. Nous mettrons l'accent sur le manque de structure de ceux-ci, et par conséquent l'impossibilité de les traiter au niveau statistique.

3e volet : Suivi du patient depuis son admission jusqu'à la fin de son séjour dans la clinique.

c) Etude de la solution projetée

Au vu du système existant et des besoins exprimés, nous proposerons dans cette troisième étape la solution envisagée, et un diagramme des flux illustrera cette proposition. Les problèmes rencontrés y seront exprimés ainsi que les nouvelles structures envisagées.

d) Analyse fonctionnelle proprement dite

L'analyse fonctionnelle :

- étudie et décrit les données utilisées
- fournit une hiérarchie dans la spécification des traitements
- étudie la coordination des traitements et
- spécifie les règles de traitements.

Afin de réaliser ces buts, nous distinguons quatre grandes étapes :

d1. DESCRIPTION DES DONNEES

Description des données selon un schéma Entités/Association.

d2. DESCRIPTION DES TRAITEMENTS

- Décomposition du projet en applications
- Décomposition des applications en phases
- Décomposition des phases en fonctions

d3. DYNAMIQUE DES TRAITEMENTS

Elle nous fournit les mécanismes ou les concepts qui permettent de représenter les conditions de déclenchement, d'exécution et d'enchaînement des fonctions entre-elles.

d4. STATIQUE DES TRAITEMENTS

Elle met en évidence les règles de traitement pour obtenir les messages de sortie à partir des messages d'entrée pour chaque fonction.

e) Analyse organique

Cette étape d'analyse organique concerne plus directement le schéma des accès aux données et la structure modulaire des traitements. Dès lors, cette analyse comprend :

- e1. Le schéma des accès possibles aux données conformément au schéma conceptuel;
- e2. Etude de la configuration physique du système tant au niveau matériel qu'au niveau du langage;
- e3. Schéma des accès nécessaires compte tenu des contraintes du système
- e4. Algorithmes de haut niveau des principales fonctions.

f) Implémentation

L'implémentation du projet concerne plus particulièrement la partie réalisation. Elle concerne :

- la programmation des différents modules;
- l'enchaînement de ceux-ci et l'intégration dans la base de données existante;

- édition des résultats de quelques tests;
- édition de quelques listings représentatifs.

g) Conclusions et perspectives

Enfin, la dernière partie est consacrée aux conclusions, où nous apprécierons les résultats obtenus par rapport à ceux escomptés, et nous fera percevoir les perspectives d'avenir de ce projet.

II. ETUDE D'OPPORTUNITE

1. BESOINS SOUHAITES PAR LA COMMISSION D'HYGIENE

La commission d'hygiène de la clinique est présidée par le médecin directeur et le bactériologiste en est son adjoint. Cette commission a pour rôle principal de contrôler et de réagir face aux infections qui surgissent dans la clinique.

Dès lors, cette commission souhaite obtenir des statistiques et des informations précises au sujet des infections suspectées.

Les propositions de cette commission se définissent en ces buts :

- a) Déterminer le bruit de fond ou le degré moyen d'infections par site et par salle en vue de repérer les unités de soins où les enquêtes épidémiologiques seraient nécessaires.
- b) Trier la liste des infections selon des critères de tri au choix.
- c) Classifier les souches d'un même germe en fonction de son profil antibiotique.
- d) Définir un fichier de germes.
- e) Définir un fichier des sites de prélèvements.

2. BESOINS SOUHAITES PAR LE MEDECIN DU SERVICE DE REANIMATION

Le service de réanimation est celui où sévit le plus d'infection. Les patients hospitalisés dans ce service sont les plus suivis, entre autre au niveau bactériologique. En effet, les tubages, les sondes vésicales, les redons sont tous des portes d'entrée aux germes et aux infections. Il est donc important de connaître le plus rapidement possible un cas d'infection, de manière à l'enrayer au plus vite. De cette constatation découlent normalement deux exigences :

- a) Détecter rapidement une infection définie comme pathologique.
- b) Adapter le plus précisément possible l'antibiothérapie ce qui signifie, fournir les histogrammes de sensibilité - résistance de chaque espèce bactérienne en vue d'une modification éventuelle des schémas antibiotiques.

3. BESOINS SOUHAITES PAR LE BACTERIOLOGISTE EPIDEMIOLOGISTE

Le rôle de l'épidémiologiste, consiste à contrôler, à déterminer, à vérifier l'évolution du taux d'infection dans les services. Il est aussi très attentif à la propreté du matériel, des instruments chirurgicaux, des salles, des chambres, des "surfaces" de contact...

Aucune donnée ne permet actuellement de contrôler l'hygiène des soins prodigués aux patients, dans les différents services. Aucune étude n'est faite quant à l'adaptation des germes à certains antibiotiques. Il s'avère cependant que l'utilisation forcée de certains antibiotiques provoquent une résistance plus grande du germe à cet antibiotique. Le bactériologiste demande une possibilité de :

- a) Contrôler et connaître les germes "poly-résistants".
- b) Fournir quotidiennement une liste d'infections nosocomiales possibles en vue du passage de ce dernier dans les unités de soins.
- c) Fournir quotidiennement une liste des "cross-infection" possible.
- d) Fournir la répartition des bactéries rencontrées selon les services et la nature des prélèvements.

III. ETUDE DE L'EXISTANT

1. ETUDE ET CONTEXTE DU PROJET AU NIVEAU DU LABORATOIRE

Le médecin prescrit une demande pour un patient. Il existe trois types de demandes : - demandes "consultants"
- demandes "hospitalisés"
- demandes "apportées"

Les demandes sont dites "consultants", si le patient arrive au secrétariat avec sa (ses) demande(s) d'analyse, et les prélèvements sont réalisés par une personne qualifiée, dans un local de prélèvement.

Les demandes sont dites "hospitalisés", si elles proviennent de médecins internes à la clinique, pour des patients hospitalisés.

Les demandes sont dites "apportées", si le médecin prescripteur vient apporter lui-même, au secrétariat, les demandes et les prélèvements pour ses patients.

Afin de mieux comprendre le cheminement suivi par une demande d'analyse, citons les quatre grandes étapes fondamentales de cette évolution et, suivant la nature des demandes, nous étudierons plus en détail l'agencement et l'endroit où ces traitements sont exécutés. Ces quatre grands traitements sont :

- 1) Encodage de la demande et prélèvement des échantillons
- 2) Réception des échantillons au dispatch du laboratoire
- 3) Dosage des analyses par station de travail et encodage des résultats
- 4) Validation médicale des résultats et édition des protocoles

Suivant le type de la demande la deuxième partie de l'étape 1 peut ne pas exister ou être traitée différemment par rapport aux autres demandes. Aussi, pour mieux comprendre l'organisation, nous étudierons l'étape 1 pour :

- 1.1. demandes "consultants" et "apportées"
- 1.2. demandes "hospitalisés"

Pour les demandes "hospitalisés", nous distinguerons :

- 1.2.1. demande non urgente d'analyse sanguine
- 1.2.2. demande non urgente d'analyse non sanguine
- 1.2.3. demande d'analyse urgente

En second lieu, nous étudierons 2, 3, et 4 qui sont communes aux différents types de prélèvements.

1.1. Encodage de la demande et prélèvement des échantillons

1.1.1. Les demandes "consultants" et "apportées"

La demande prescrite par le médecin est encodée au secrétariat du laboratoire, où on vérifie l'identité du patient.

Si le patient existe dans la D.B., la demande est encodée et un numéro de demande est attribué en même temps que des étiquettes sont éditées :

- des étiquettes iront sur la demande;
- des étiquettes iront sur les échantillons prélevés.

Si le patient n'existe pas, il est au préalable créé et introduit dans la D.B., avant d'encoder, d'attribuer un numéro de demande, et d'éditer des étiquettes.

Si le prélèvement n'a pas encore été réalisé, le patient passe au local de prélèvement où la personne qualifiée consulte la demande, prélève la quantité nécessaire dans les tubes adéquats et étiquette ces échantillons. Si le prélèvement a déjà eu lieu (demandes apportées ou prélèvement déjà réalisé), on colle les étiquettes sur les échantillons.

Les échantillons étiquetés, et les demandes qui y correspondent sont descendus au dispatch du laboratoire.

1.1.2. Les demandes "hospitalisés"

1.1.2.1. Demande d'analyse de sang non urgente :

Les demandes sont descendues à la réception du laboratoire, la veille du dosage. Ces demandes sont descendues avant 19 h. Le soir, ou le lendemain matin, une personne encode les demandes. Un numéro de demande est attribué directement, tandis que des étiquettes sont éditées. Les étiquettes "demande" sont collées sur la demande. Les étiquettes destinées aux échantillons sont aggrafées aux demandes. Le jour du dosage, la personne chargée de réaliser le prélèvement, emporte les demandes encodées et les étiquettes. Le prélèvement a lieu dans les services et les tubes des échantillons sont étiquetés. Les demandes et les échantillons prélevés étiquetés descendent ensuite au dispatch du laboratoire, où les échantillons seront réceptionnés en regard de la demande.

1.1.2.2. Demande d'analyse sur liquides divers non urgente

Etant donné la nature des prélèvements (selles, urines, frottis), ceux-ci sont récoltés par une infirmière du service et sont descendus au dispatch du laboratoire en même temps que la demande. On encode ces demandes, on édite les étiquettes, on réceptionne les échantillons, et on les étiquette.

1.1.2.3. Demande d'analyse urgente

Soit :

- La demande arrive au dispatch du laboratoire où elle est encodée. Une personne qualifiée s'empare de la demande et va prélever. Les échantillons prélevés sont ensuite réceptionnés et étiquetés.

- La demande et le prélèvement sont descendus au service de réception du laboratoire, où la demande est encodée. Un numéro de demande est attribué, tandis que des étiquettes sont éditées et les échantillons réceptionnés.

1.2. Réception des échantillons au dispatch du laboratoire

Les tubes réceptionnés sont triés par stations de travail (bouchons des échantillons différents). Les laborantines viennent y chercher les échantillons pour doser les analyses qui leur sont impartites. En fin de journée, les demandes accumulées au dispatch du laboratoire sont renvoyées au secrétariat où elles sont classées.

1.3. Dosage des analyses par station de travail

Les stations de travail se divisent en deux groupes :

- Les stations manuelles : les analyses sont dosées manuellement et les résultats sont ensuite encodés par les techniciennes.
- Les stations automatiques : les analyses sont effectuées par un analyseur qui est connecté au système informatique; les résultats ainsi transmis automatiquement peuvent être validés techniquement avant d'être visualisés par les médecins valideurs du laboratoire.

1.4. Validation médicale des résultats et édition des protocoles

Les résultats répondus sont validés médicalement et, généralement en fin de journée, les protocoles sont édités et envoyés au(x) médecin(s) prescripteur(s).

2. ETUDE ET CONTEXTE DU PROJET EN BACTERIOLOGIE

Dans l'étude du contexte du laboratoire et plus particulièrement au niveau de la bactériologie, nous pouvons dire que le suivi général des demandes d'analyses est le même. De la même manière, nous pouvons distinguer des demandes :

- de type "consultant" urgentes ou non
- de type "hospitalisés" urgentes ou non

La différence essentielle ne réside pas dans la manière de traiter la demande en tant que telle, mais ce sont les informations qu'elle véhicule que nous devons mieux préciser. Aussi, parmi les quatre grands traitements traités ci-dessus, nous ne développerons que les étapes 1.1 et 1.3. En effet, les étapes 1.2 et 1.4 étant tout à fait identiques dans un contexte comme dans un autre, nous décidons de ne pas décrire une seconde fois ces informations.

2.0. Introduction

Afin de mieux comprendre le suivi d'une demande bactériologique depuis sa réception au laboratoire jusqu'à l'édition des résultats sur protocoles, nous décrirons en trois étapes la technique généralement utilisée et les informations qui y sont véhiculées.

Notons dès à présent, que la notion de site de prélèvement est caractéristique en bactériologie. Par définition, il s'agit de la localisation anatomique du patient où a été prélevé l'échantillon. Dans certaines

publications cette localisation est appelée origine du prélèvement ou plus couramment body-site.

1ère étape : examen direct

Quand un prélèvement arrive au laboratoire, l'analyse bactériologique commence par un EXAMEN DIRECT. L'examen direct consiste à visualiser sous microscope optique le contenu de l'échantillon. La physiologie des cellules, la présence de cristaux, la coloration gram, la forme des germes rencontrés, la présence de parasites sont des exemples de résultats possibles.

2e étape : culture

Parallèlement à l'examen direct, la technicienne met en route la CULTURE sur milieu aérobie ou anaérobie. C'est-à-dire que l'on ensemence un milieu riche, propice à la multiplication des germes, l'échantillon à analyser. Le résultat est négatif si rien ne s'est multiplié, il est positif si un ou plusieurs germes se sont multipliés dans le milieu de culture.

Une première approximation (qualitative ou quantitative) peut être répondeue. De la même manière, on peut évaluer grossièrement l'identité du (des) germe(s) mis en évidence.

3e étape : antibiogramme

En même temps que la culture, on met en route le test de sensibilité du (des) germes à certains antibiotiques choisis. Cette analyse s'appelle l'ANTIBIOGRAMME et le résultat est de type sensible si le germe a été éliminé par la présence de l'antibiotique ou résistant si cet antibiotique n'a exercé aucune influence sur la multiplication du germe.

2.1. Encodage de la demande

La demande prescrite par le médecin est encodée au secrétariat où on vérifie l'identité du patient. Sans trop détailler, nous dirons qu'une demande de bactériologie regroupe plusieurs analyses différentes :

- l'origine du prélèvement
- l'examen direct
- la culture
- antibiogramme

Si le patient n'existe pas dans la base de données on le crée, sinon on encode pour la demande ces différentes analyses. Si le site de prélèvement intrinsèque au code de l'analyse doit être précisé, celui-ci l'est à condition que l'encodeuse le précise en texte libre (Ex : tubage poumon gauche).

2.2. Réception des échantillons au dispatch du laboratoire

Les tubes réceptionnés sont triés par stations de travail (bouchons des échantillons différents). Les laborantines viennent y chercher les échantillons pour doser les analyses qui leur sont impartites. En fin de journée, les demandes accumulées au dispatch du laboratoire sont renvoyées au secrétariat où elles sont classées.

2.3. Dosage des analyses par station de travail

Le dosage des analyses de type bactériologique sont différentes des autres analyses. Outre les sites de prélèvements, on distingue globalement trois analyses et par conséquent trois résultats différents pour une analyse bactériologique. Ces analyses ne sont pas toutes répondues en même temps, étant donné la technique d'analyse. Celle-ci peut se répartir en quatre étapes.

Soit l'exemple d'une demande de bactériologie urinaire.

1. La technicienne procède à l'examen direct sous microscope de l'échantillon d'urine étalé sur lame de verre et introduit les résultats des analyses en texte libre.
2. On ensemence dans la même journée une micro-goutte d'urine sur un milieu riche afin d'être dans les meilleures conditions possibles pour faire "pousser" les éventuels germes présents.
3. Le jour suivant, on lit la culture. Si elle est négative, on répond négatif (en texte libre) à la culture et l'analyse est clôturée. Dans ce cas, la technicienne doit supprimer pour cette demande, les analyses antibiogrammes prescrites.
Si elle est positive, la demande n'est pas clôturée et on procède le même jour J :
 - à la numération des germes;
 - à l'identification des germes (à partir, le cas échéant de galeries d'identification);
 - à la mise en route de l'antibiogramme.La numération, ainsi que les germes rencontrés sont répondus en texte libre.
Dans le cas d'une identification incomplète, la demande pourra être corrigée lorsque le germe sera plus précisément identifié.
4. Deux jours après la réception de l'échantillon, on lit le résultat de l'antibiogramme. Ce résultat n'est pas répondu en texte libre. En effet, la grille des antibiotiques pour cet antibiogramme est affichée à l'écran et la technicienne répond S(ensible), I(ntermédiaire), R(ésistant) pour ce germe et cet antibiotique.

2.4. Validation médicale des résultats et édition des ----- protocoles -----

Voir annexe : dynamique des traitements

Voir annexe : exemple de protocoles

3. ETUDE ET CONTEXTE DU PROJET AU NIVEAU DE L'ADMISSION

3.1. Notions d'admission et de séjour -----

On entend par admission d'un patient, une durée d'hospitalisation dans la clinique d'au moins un jour. Cette hospitalisation fait l'objet de la création d'un nouveau numéro dit numéro d'admission, qui suit le patient tout au long de son hospitalisation. Ce numéro sera nouvellement attribué lorsque le patient reviendra dans la clinique contrairement au numéro du dossier patient créé lors de son inscription dans la base de données et qui est fixé à vie.

A ce numéro d'admission correspond la notion de séjour. On définit un séjour comme étant une nouvelle attribution au patient d'un lit dans un service. De cette manière, lors d'un transfert d'un lit dans un autre, ou d'un service vers un autre service, un nouveau séjour est créé.

A ce séjour, s'adjoignent les renseignements tels que :

- type de séjour
- médecin responsable
- service infirmier
- date d'admission
- date du début du séjour dans ce service
- date de fin du séjour dans ce service
- ...

3.2. Suivi d'une demande d'hospitalisation -----

3.2.1. Admission -----

On distingue deux types de demandes d'hospitalisation :

- Demande d'hospitalisation non urgente
- Demande d'hospitalisation urgente

La demande d'hospitalisation est dite non urgente, si le patient arrive au secrétariat de l'admission, avec la demande d'hospitalisation prescrite par un médecin.

La demande est urgente, si elle émane d'un service de la clinique où le patient vient d'être hospitalisé. Afin de mieux comprendre l'enchaînement logique des traitements

référons-nous au diagramme des flux.

Voir annexe : Diagramme des flux d'une admission

3.2.1.1. Admission non urgente

Au secrétariat de l'admission, la secrétaire vérifie l'existence du patient dans la base de données, à partir de son numéro de dossier. Si celui-ci existe, elle reçoit en écho, son signalétique et le numéro de son dossier qui l'identifie, sinon elle encode les renseignements qui le concernent et un numéro lui est attribué. En même temps, il y a une vérification de l'existence du médecin dans la base de données. Si celui-ci n'existe pas, elle le crée.

Chaque matin et chaque après-midi, la secrétaire édite la liste de lits libres par service. Elle consulte la liste et la demande du patient, auquel elle attribue un lit, en essayant de tenir compte de ses souhaits quant au type de chambre. Si aucun lit n'est libre dans ce service, soit elle refuse la demande soit elle l'affecte dans un service moins complet, en attendant la vacance d'une place dans le service demandé.

Quand un lit a été trouvé, la secrétaire opère au transfert de type "entrée", c'est-à-dire qu'un numéro d'admission va être attribué au patient pour toute sa durée d'hospitalisation et un séjour va être créé pour le service dans lequel il entre.

Parallèlement, des étiquettes signalétiques du patient sont éditées à destination du service, et quatre avis d'hospitalisation sont destinés au médecin, au service, et à l'administration de la clinique.

3.2.1.2. Admission urgente

Le service qui reçoit en urgence un patient, le signale au secrétariat de l'admission et lui précise le lit, le type de chambre occupée. La secrétaire vérifie l'identité du patient, et le crée s'il n'existe pas encore dans la base de données. Par le transfert qu'elle exécute, un numéro d'admission et un séjour sont créés pour ce patient. De la même manière, des étiquettes et des avis d'hospitalisation sont édités.

3.2.2. Autres transferts

3.2.2.1. Transfert d'un lit A vers un lit B

Quand un patient change de lit (ou de chambre), cela doit être signalé au secrétariat, afin de mettre à jour les fichiers de la base de données et de créer un nouveau séjour pour ce patient. Des étiquettes du patient sont éditées et un formulaire de transfert est destiné au service.

3.2.2.2. Prolongation séjour

Un patient peut voir son séjour à la clinique prolongé. Le service où se trouve ce patient envoie au secrétariat une demande de prolongation, de manière à mettre à jour les informations dans la base de données comme, par exemple, la date de sortie du patient. Un avis de prolongation est édité pour la mutuelle où un médecin conseil validera ou non cette prolongation de séjour. La réponse à cet avis de prolongation revient au service de l'admission qui en tiendra compte lors de la facturation.

3.2.2.3. Sortie du patient

A la sortie de l'hospitalisation, le patient passe au secrétariat où la secrétaire enregistre la fin du séjour et clôture le séjour du patient. La base de données est mise à jour et un avis de sortie est édité à la mutuelle du patient.

IV. ETUDE ET DESCRIPTION DE LA SOLUTION PROJETEE

0. INTRODUCTION

Le projet de surveillance épidémiologique résulte d'une confrontation de données administratives et médicales des patients hospitalisés.

Les données administratives dont nous avons besoin sont :

- la date d'admission
- le numéro d'hospitalisation
- la localisation du patient dans l'hôpital
- le suivi du patient dans les différents services au cours de son séjour

Ayant étudié et décrit le suivi d'une admission d'un patient dans l'hôpital, nous savons que nous pouvons connaître à tout moment les informations concernant son séjour à partir de son numéro de dossier et de son numéro d'admission. Dès lors, le projet a à sa disposition toutes les données administratives voulues, et rien ne devra être modifié au niveau de ces informations.

Les données médicales et plus particulièrement les résultats bactériologiques provenant du laboratoire sont :

- site de prélèvement
- l'identification précise des germes
- la numérotation quantitative/qualitative de chaque germe
- les résultats de sensibilité/résistance aux antibiotiques

Ces informations sont connues dans la base de données mais ne sont pas directement exploitables. En effet, comme nous l'avons expliqué dans le chapitre qui concerne la critique de l'existant, toutes ces données, hormis les résultats aux antibiotiques, sont codés en texte libre.

Pour réaliser notre objectif, nous étudierons de nouvelles structures de données et les adaptations que celles-ci ont impliquées.

L'analyse fonctionnelle et organique qui seront réalisées, se rapporteront uniquement à l'application épidémiologie, dont la description de la solution projetée se trouve ci-après.

En effet, l'application bactériologique existe, et pour être compatible avec la nouvelle application épidémiologie, il a fallu adapter de nouveaux concepts, créer de nouvelles analyses, mais l'architecture principale de l'application n'a pas dû être remise en cause.

Aussi, nous nous sommes restreints à décrire les modifications qui ont été apportées pour la première application et à analyser précisément la seconde.

Toutefois, il est important de signaler que la nouvelle application bactériologie a nécessité de nombreuses heures de travail et a demandé de revoir la programmation d'environ trente modules dont dix ont été refaits complètement.

1. APPLICATION BACTERIOLOGIE

1.1. Nouvelles structures des résultats de bactériologie et

leurs implications

Nous rappelons qu'une demande de bactériologie prescrite par un médecin pour un patient se compose de plusieurs analyses. Le signalétique de chaque analyse possède entre autre, un code INAMI, un libellé francophone et néerlandophone, des unités qui la caractérisent, mais aussi, au niveau de la demande, un statut.

```
Ex : Statut 1 : analyse demandée
           3 : analyse réceptionnée
           4 : analyse passée sur liste de travail
           5 : analyse supprimée
           7 : analyse répondue
           8 : analyse annulée
           9 : analyse validée
          10 : analyse passée sur protocole
          11 : analyse archivée
```

Les résultats que nous devons pouvoir exploiter sont :

- les sites de prélèvements
- les résultats de la culture en tant qu'identification précise du germe et de sa numération
- les résultats aux antibiotiques testés

Envisager de créer une seule analyse globale multiple
qui reprendrait les analyses

- origine du prélèvement
- examen direct du prélèvement
- culture
- l'antibiogramme

était une possibilité.

Mais quel statut donner à cette analyse ?
 Quel type de résultat donner à cette analyse ?
 Mais quel code INAMI lui attribuer ?

De plus, la technique en bactériologie est telle qu'il peut se passer de 48 heures à plusieurs jours, voire plusieurs semaines entre la réception des échantillons et la clôture du dosage des analyses.

Nous avons alors envisagé de créer de nouvelles analyses avec de nouveaux types de résultats. Jusqu'à ce jour, on distinguait comme type de résultats :

```
type résultat 1 : type numérique
              " 2 : type dilution
              " 3 : type Positif, Négatif, Douteux
              " 7 : type RAST (se réfère à la table des aller-
                  gènes)
```


type résultat 8 : type antibiogramme (se réfère à la table
des antibiogrammes)
" 9 : type expression
" 10 : type texte libre

Nous avons ajouté deux autres types :

type résultat 6 : type "bactério" qui concerne les analyses
cultures
" 4 : type body-site qui concerne les analyses
origines

L'analyse de type body-site se réfère à un nouveau
fichier des sites de prélèvement. L'analyse culture se
réfère à un nouveau fichier de germes.

Ces analyses créées, ces types de résultat implémentés,
les fichiers décrits, examinons la nouvelle application
bactériologique et ses implications au niveau :

- 1) Encodage de la demande et prélèvement des échantillons
- 2) Réception des échantillons au dispatch du laboratoire
- 3) Dosage des analyses par station de travail
- 4) Validation médicale des résultats et édition des
protocoles

1.1.1. Encodage de la demande et prélèvement des échantillons

La demande prescrite par le médecin est encodée. Si le
patient n'existe pas dans la base de données, on le crée,
sinon un numéro de demande est directement attribué et
des étiquettes sont éditées. Si une analyse de type
bactériologique est encodée pour cette demande, l'écho du
site de prélèvement intrinsèque à l'analyse peut être
affichée à l'écran. Si une identification plus précise
du site de prélèvement est connu, la secrétaire encode
pour l'analyse le code de celui-ci. Si ce code n'existe
pas encore dans la table des sites de prélèvement, soit
elle le précise en texte libre, soit elle créera un
nouveau site de prélèvement dans la table après avoir
reçu l'accord du bactériologiste. Si aucune identifica-
tion plus précise n'est connue, le body-site intrinsèque
à l'analyse est conservé.

Notons que pour la facilité d'encodage de ces body-sites,
nous avons prévu une recherche par famille et par sous-
famille. Après avoir choisi la famille ou sous-famille
du body-site, la secrétaire reçoit en écho à l'écran la
liste des codes des body-sites qui y figurent.

Voir annexe : Nouvel encodage de la demande

1.1.2. Réception des échantillons au dispatch du ----- laboratoire -----

Dans cette phase rien n'a été modifié.

1.1.3. Dosage des analyses par station de travail -----

La technique d'analyse est inchangée. Par contre, c'est l'introduction de résultats qui est modifiée. Afin de mieux comprendre et de comparer plus aisément l'introduction des résultats avant et après changements des structures, référons-nous au même exemple choisi lors de la critique de l'existant.

1) La technicienne procède à l'examen direct sous microscope de l'échantillon d'urine et introduit le résultat de cette analyse en texte libre.

2) Dans la même journée, elle ensemence une micro-goutte d'urine sur milieu riche.

3) Le jour suivant, elle lit la culture.

- Si elle est négative, elle répond 'N' à la culture et l(es) antibiogramme(s) de la demande sont supprimés automatiquement.

- Si elle est positive, elle répond 'P' à la culture et répond à la numération (qualitative et quantitative) de la (des) souche(s) identifiées. Elle répond à l'identification des souches identifiées par un code de germe.

Si ce germe n'est pas encore identifié de manière précise, un code de germe imprécis aura été créé mais ne sera jamais pris en compte lors des traitements ultérieurs sur les données bactériologiques. Elle met en route l'antibiogramme.

4) Elle introduit le résultat de l'antibiogramme.

Dynamique des traitements

Voir annexe : Dynamique des traitements du nouvel encodage des résultats.

1.1.4. Validation médicale des résultats et édition des ----- protocoles -----

Les résultats répondus sont validés à l'écran par le médecin et les protocoles sont édités en fin de journée.

Voir annexe : Nouvelle édition des protocoles.

2. APPLICATION EPIDEMIOLOGIE

2.1. Objectifs à réaliser

Afin de répondre aux besoins exprimés par les utilisateurs, nous distinguerons deux grands types de traitements :

- Le traitement journalier
- Les traitements "a posteriori" sur une période de temps définie par les utilisateurs

Le traitement journalier a pour but de proposer journalièrement au médecin hygiéniste, en vue de son passage dans les unités de soins :

- 1) Une liste des nouveaux cas d'infections pathologiques
- 2) Une liste des candidats possibles de "cross-infection"

Les traitements "a posteriori" consistent :

- 1) A l'édition de listes des résultats bactériologiques selon certains critères de tri, de sélection choisis.
- 2) En différents relevés statistiques des données bactériologiques mémorisées.

Dans le cadre limité de ce travail de fin d'études, nous retiendrons les statistiques et listes suivantes :

- a) Liste des patients infectés triés par germe/service
- b) Liste des patients infectés triés par service/germe
- c) Liste de répartition des germes par service

A noter, que les listes journalières sont surtout utiles au médecin, pour réagir ponctuellement à de nouvelles infections et ainsi éviter toute contamination d'un patient par un autre. Par opposition, les autres listes se réfèrent à des périodes de temps plus longues et permettent de mettre en exergue toute infection nosocomiale liée à l'hygiène hospitalière proprement dite. Ex : contamination par matériel, mains souillées, personnel infirmier, sonde vésicale contaminée, etc.

2.2. Définition et concepts

2.2.0. Notions body-sites "master/slave"

La notion de famille et de sous-famille de body-site permet de classer les sites de prélèvements. Un body-site "slave" par rapport à un body-site "master" est un site de prélèvement qui appartient à la même famille, mais qui possède un degré de précision supplémentaire. Ainsi, par exemple, le body-site de famille PUS sera dit "master" par rapport au body-site PUP (pour PUs Plaies) qui en est un "slave".

2.2.1. Pathogénicité d'une infection

La pathogénicité d'une infection s'exprime par la présence d'un germe (dit pathogène) dans un site de prélèvement précis ou non. D'autres caractéristiques telles que la numération du germe dans le body-site, ou la résistance/sensibilité du germe à certains antibiotiques permettent d'affiner cette relation.

2.2.2. Cross-infection

Une cross-infection est une infection causée par un germe dit germe cible. Celui-ci peut être caractérisé par une numération, un body-site, une résistance/sensibilité à certains antibiotiques. Cette infection est dite croisée ("cross-infection") car un patient porteur du germe peut contaminer facilement un autre patient non encore porteur.

2.2.3. Délai de non redondance

Délai exprimé en jours tel qu'une occurrence d'un germe dans un body-site chez un patient n'est significative que si la période de temps qui la sépare de la dernière occurrence de ce germe dans ce body-site pour ce patient est supérieure ou égale à ce délai.

2.2.4. Résultat non redondant

Un résultat bactériologique positif exprimé par la présence d'un germe dans un body-site chez un patient n'est pas redondant si :

- la période de temps qui le sépare de la dernière occurrence de ce germe, dans le même body-site "master", pour le même patient est \geq délai de non redondance.
- si ce résultat est séparé de la dernière occurrence du même germe, dans le même body-site "master", chez le même patient par un résultat négatif.

On dira que le résultat non redondant est une "mère" par opposition à un résultat redondant appelé nouvelle itération de cette "mère".

2.2.5. Résultat bactériologique journalier à traiter

Tout résultat bactériologique comprenant un résultat pour l'analyse culture sera pris en compte si :

- cette culture est négative
- cette culture est positive à condition que le germe soit identifié de manière précise.

(Remarque : les cultures contaminées sont considérées comme négatives.)

2.3. Traitement journalier

Journellement, chaque résultat bactériologique encodé fera l'objet d'un traitement journalier scindé en trois parties :

- 1) Test de redondance
- 2) Test de pathogénicité
- 3) Test si cible

2.3.1. Test de redondance

Ce test est réalisé afin de définir s'il s'agit d'une "mère" ou d'une nouvelle itération d'une mère identifiée. Cette nouvelle "mère" sera définie comme étant la dernière "mère" connue.

Le délai de redondance considéré dans ce test sera lu dans la table des germes qu'on rencontre dans ce body-site. Si ce délai est imprécisé à ce niveau, un paramètre interne de défaut spécifiera cette donnée. Remarquons, que si le délai de redondance vient à être modifié toutes les occurrences qui étaient considérées comme "mère" ne le sont plus nécessairement. Il faut donc réattribuer la dernière "mère" connue.

2.3.2. Test de pathogénicité

Le résultat bactériologique du jour est comparé à la table des germes pathogènes. Ce résultat sera considéré comme pathogène si :

- il y a similitude entre les body-sites et le germe
- il y a similitude au niveau de la numération (si elle est précisée)
- il y a similitude a plus de X % au niveau des antibiogrammes (si ceux-ci sont précisés).

Le facteur X est un paramètre interne à l'application épidémiologie et modifiable par l'utilisateur.

Le résultat sera édité si le patient concerné est toujours hospitalisé et si la dernière "mère" connue n'a pas fait l'objet d'une édition similaire.

Etant donné la difficulté de définir le délai de non redondance pour un germe et, dans le but de ne pas oublier l'édition d'un cas d'infection pathogène grave, il sera précisé dans la table des germes pathogènes, un délai d'édition d'une infection.

2.3.3. Test si cible

a) Comparaison table des germes cibles

Chaque résultat bactériologique du jour est comparé à la table des germes cibles. Ce résultat sera considéré comme cible si :

- il y a similitude entre body-site et le germe
- il y a similitude au niveau de la numération (si précisée)
- il y a similitude à plus de X % au niveau des résultats des antibiotiques testés (si précisés)

Le facteur X est un facteur interne au système pour l'application épidémiologie.

Si le patient concerné est hospitalisé depuis au moins 48 heures et si la dernière "mère" n'a pas déjà fait l'objet d'une édition sur la liste des "cross-infection", l'étape b) sera entamée pour ce résultat bactério.

b) Recherche des candidats

Les résultats bactériologiques sélectionnés dans la partie a) sont comparés à tous les résultats bactériologiques obtenus précédemment. Les patients correspondants seront édités comme candidat "contaminant" s'ils sont hospitalisés depuis au moins 48 heures et s'ils sont toujours présents dans la clinique au moment de la sortie de la liste.

Afin de ne pas proposer sur la liste des cas de "cross-infection" connus, un délai de recherche sera précisé dans la table des germes cibles.

Les sept critères de recherche des candidats "contaminant" sont :

- 1) Le germe est-il le même ?
- 2) Le patient est-il le même ?
- 3) Le patient est-il hospitalisé depuis au moins 48 heures ?
- 4) Le patient est-il toujours hospitalisé ?
- 5) L'antibiogramme (résultats aux antibiotiques) est-il le même à X % ?
- 6) Le patient est-il hospitalisé depuis longtemps ?
- 7) La salle où séjournent les patients est-elle la même ?

La réponse à ces critères est Oui/Non. L'option à prendre dans chaque cas peut être visualisée par une table de décision. Cette table de décision se traduit normalement par 64 possibilités. Nous avons pu la simplifier :

SITE ID.	GERME ID.	PAT ID.	HOSP 48H?	HOSP ACT.	ABG ID. X %	HOSP LONG.	SALLE ID.	
=	N	=	=	=	=	=	=	E
=	O	O	=	=	=	=	=	E
=	O	N	N	=	=	=	=	E
=	O	N	O	N	=	=	=	E
=	O	N	O	O	=	O	=	E
=	O	N	O	O	O	N	O	B.C+A
=	O	N	O	O	O	N	N	B.C+AA
=	O	N	O	O	N	N	O	B.C+A ABG*
=	O	N	O	O	N	N	N	B.C+AA ABG*

LEXIQUE :

"O" : Oui
 "N" : Non
 "=" : Ne pas prendre en compte
 "B.C" : Bon Candidat
 "A" : Alarme
 "AA" : Grande alarme
 "E" : Eliminer

2.3.4. Format des listes journalières

Voir annexe : Liste des infection pathologiques
 Liste des candidats "cross-infection"

2.4. Traitement "a posteriori"

A tout moment, l'utilisateur peut demander l'édition des listes, ou l'affichage de certaines statistiques.

Parmi les statistiques et listes disponibles, nous détaillerons celles qui nous ont parues les plus utiles. Pour chacune d'elles, nous procéderons de la même manière :

- description de la liste
- intérêt direct de la liste
- la table de décision sous-jacente au traitement de la liste
- le format de la liste

2.4.1. Listes des patients infectés triés par germe et par service

a) Description

A partir de la période précisée par l'utilisateur, du germe et du service, on recherche tous les patients qui ont contracté ce germe durant cette période. Ces cas d'infection sont tels qu'aucun patient ne se retrouvera

deux fois sur la même liste à moins qu'il ne s'agisse de deux occurrences du même germe. En d'autres termes cela signifie que tous les cas relevés ne seront pas redondants.

b) Intérêt de la liste

Cette liste présente comme intérêt majeur de recenser de manière nominative les cas d'infection afin de déterminer plus précisément les types de patients infectés.

c) Table de décision

La table de décision concernée dans cette recherche de candidats est la suivante :

SITE ID	GERME ID	PAT ID	PERIODE OK	SERVICE OK	NON RED.	
=	N	=	=	=	=	E
=	O	=	N	=	=	E
=	O	=	O	N	=	E
=	O	=	O	O	N	E
=	O	=	O	O	O	Pris en compte

LEXIQUE :

"O" : Oui

"N" : Non

"=" : Ne pas prendre en compte

"E" : Eliminer

Remarque : Si on introduit '*' pour le germe ou le service, on ne tient pas compte de l'élément correspondant. Il en sera de même pour les autres listes.

d) Format de la liste

Voir annexe : Liste des patients infectés triés par germe et par service

2.4.2. Liste des patients infectés triés par service et ----- par germe -----

Même sélection que 2.4.1. mais le tri est différent.

2.4.3. Liste de répartition des germes par service

a) Description

A partir de la période précisée par l'utilisateur, du (des) germes et du (des) services, on détermine les cas isolés non redondants et leur nombre.

b) Intérêt de la liste

Cette liste présente comme intérêt majeur de recenser par service le nombre de germes rencontrés. Cette statistique a pour intérêt de préciser le service où sévit le plus d'infection et le type de germe qui a été la cause de ces infections.

c) Table de décision

Idem 2.4.1.

d) Format de la liste

Voir annexe : Liste de répartition des germes par service

2.4.4. Taux d'infection par service

a) Description

A partir de la période précisée par l'utilisateur et du (des) service(s), on détermine les prélèvements recueillis sur cette période, leur nombre et la fréquence des infections.

b) Intérêt de la liste

Cette liste présente comme avantage de pouvoir comparer deux à deux des services au niveau du taux d'infection global par rapport au nombre de prélèvements réalisés. De cette façon, on peut spécifier le service où des enquêtes épidémiologiques seraient utiles voire nécessaires.

c) Table de décision

SITE ID	GERME ID	PAT ID	PERIODE OK	SERVICE OK	NON RED.	
=	=	=	N	=	=	E
=	=	=	O	N	=	E
=	=	=	O	O	N	E
=	=	=	O	O	O	Pris en compte

d) Format de la liste

Voir annexe : Taux d'infection par service

2.4.5. Taux d'infection par body-site et par service

a) Description

Idem que 2.4.4. en précisant les germes rencontrés.

b) Intérêt de la liste

Etudier le taux d'infection en précisant les germes rencontrés. Pouvoir étudier l'évolution du taux d'infection en fonction du temps.

c) Table de décision

B-S ID	GERME ID	PAT ID	PERIODE OK	SERVICE OK	NON RED.	
N	=	=	=	=	=	E
O	=	=	N	=	=	E
O	=	=	O	N	=	E
O	=	=	O	O	N	E
O	=	=	O	O	O	Prise en compte

d) Format de la liste

Voir annexe : Taux d'infection par body-site et par service

2.4.6. Histogramme de sensibilité d'un germe

a) Description

L'utilisateur désigne par son code le germe pour lequel il désire connaître l'histogramme de sensibilité et la période sur laquelle il désire effectuer la recherche. De la même façon il désigne les codes des antibiotiques qu'il désire tester pour ce profil.

On étudie alors sur cette période précisée, tous les résultats bactériologiques positifs et on calcule en pourcentage le nombre de cas où ce germe a été sensible à tel ou tel antibiotique. Notons que pour affiner le pourcentage, nous avons choisi de prendre en considération tous les résultats redondants et non redondants.

b) Intérêt de la liste

b) Intérêt de la liste

L'intérêt principal de la liste réside dans l'aide à la prescription des antibiotiques aux patients infectés. A partir du germe qui détermine l'infection, on établit le profil de celui-ci afin d'enrayer l'infection le plus rapidement possible avec un antibiotique efficace pour le germe.

L'autre pôle d'intérêt non moins intéressant est la comparaison de la sensibilité du même germe au même antibiotique pour des patients de type ambulants ou hospitalisés. Une légère différence pourrait traduire une certaine résistance accrue pour les germes rencontrés et traités dans la clinique due à l'utilisation prépondérante de certains antibiotiques.

c) Table de décision

SITE ID	GERME ID	PAT ID	ABT ID	PATIENT AMB.	PATIENT HOS.	
=	N	=	=	=	=	E
=	O	=	N	=	=	E
=	O	=	O	O	.	NBAMB
=	O	=	O	.	O	NBHOS

LEXIQUE :

"O" : Oui
"N" : Non
"=" : Ne pas prendre en compte
NBAMB : Statistique pour patient ambulant
NBHOS : Statistique pour patient hospitalisé
"E" : Eliminer

d) Format de la liste

Voir annexe : Histogramme de sensibilité d'un germe

2.4.7. Polyrésistance des germes aux antibiotiques

a) Description

Cette statistique vise à connaître le caractère poly-résistant de certains germes à plusieurs antibiotiques choisis par l'utilisateur durant une période déterminée. Les germes rencontrés sur la période définie, qui sont résistants à tous ces antibiotiques, sont recensés.

b) Intérêt de la liste

La liste des germes poly-résistants est importante pour l'épidémiologiste. En effet, un germe poly-résistant est

considéré comme "dangereux" et difficile à enrayer, étant donné sa faible sensibilité aux antibiotiques couramment utilisés. Leur durée de vie est généralement assez grande, ce qui laisse prévoir de fortes contaminations.

Ce sont typiquement ces germes qui seront considérés comme germes cibles.

c) Table de décision

SITE	GERME	PAT	PER.	SERV.	NON RED.	ANTIB.	
=	=	=	N	=	=	=	E
=	=	=	O	=	N	=	E
=	=	=	O	=	O	N	E
=	=	=	O	=	O	O	Pris en compte

d) Format de la liste

Voir annexe : Poly-résistance des germes aux antibiotiques

V. ANALYSE FONCTIONNELLE

1. DESCRIPTION DES DONNEES

1.1. Méthodologie

La représentation des données prend en compte trois concepts :

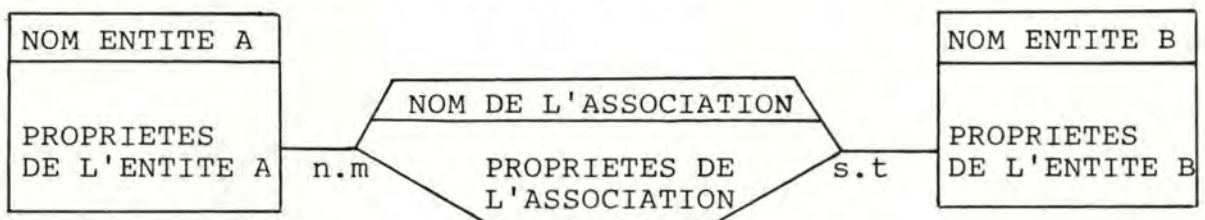
- concept d'entité
- concept d'association
- concept d'attribut

Une entité représente un objet du réel perçu. Cet objet est doté de propriétés ou attributs de l'entité.

Entre les objets, on peut trouver des relations. Une association entre deux entités peut apporter une ou plusieurs propriétés ou couple d'entités concernées. Une association se caractérise par des connectivités, c'est-à-dire le nombre minimum et maximum d'occurrences de l'entité extrémité associée à l'entité origine de la relation.

Pour être tout à fait complète, il faut mentionner à cette modélisation des informations, la notion de contraintes d'intégrités. Celles-ci sont des propriétés non représentées par les concepts de base du modèle, qui doivent satisfaire les informations appartenant à la mémoire du S.I.

La représentation schématique du modèle Entité/Association est la suivante :



R : A \rightarrow B

n = nombre minimum d'occurrence de l'entité B associée à une occurrence de l'entité A.

m = nombre maximum d'occurrence de l'entité B associée à une occurrence de l'entité A.

R : A \rightarrow B

s = nombre minimum d'occurrence de l'entité A associée à une occurrence de l'entité B.

t = nombre maximum d'occurrence de l'entité A associée à une occurrence de l'entité B.

1.2. Modèle ENTITE-ASSOCIATION

Le schéma ci-joint, en annexe, est complété par la description des contraintes d'intégrité et de la liste des propriétés des entités et des associations qui le constitue.

1.3. Liste des propriétés par Entités et Associations

1.3.1. Les entités

1.3.1.1. Entité PATIENT

Numéro-patient

Définition : Numéro de dossier du patient qui l'identifie de manière univoque dans une organisation logique.

Format : 7A*2

Date-naissance-précision

Définition : Valeur qui spécifie la précision de la date de naissance du patient.

Valeurs possibles :

0 = date de naissance connue

1 = jour de naissance inconnu

2 = jour et mois de naissance inconnus

3 = date de naissance inconnue

Format : I*2

Date-naissance-patient

Définition : Date de naissance du patient exprimée sous le format JJ-MM-AA(AA)

Format : I*2

Nom-patient

Définition : Nom de naissance du patient (de jeune fille éventuellement)

Format : 15A*1

Nom-époux-patient

Définition : Nom d'époux de la patient

Format : 15A*1

Prénom-patient

Définition : Premier prénom du patient

Format : 12A*1

Initiales-patient

Définition : Initiales du patient sous le format
X.X.X.

Format : 6A*1

Sexe-patient

Définition : Code spécifiant le sexe du patient
Valeurs possibles : M = masculin
V = féminin

Format : 1A*1

Numéro-rue-patient

Définition : Numéro et rue du domicile du patient

Format : 28A*1

Commune-patient

Définition : Valeur de référence de la table des
communes

Format : I*2

Libellé-commune-patient

Définition : Localité du domicile du patient

Format : 18A*1

Code-postal-patient

Définition : Code postal de la localité où habite le
patient

Format : I*2

Téléphone-patient

Définition : Numéro de téléphone du patient

Format : 7A*1

Localité-téléphone

Définition : Localisation où on peut joindre par téléphone le patient

Valeurs possibles : 0 = inconnu
1 = à la maison
2 = au travail
3 = ailleurs

Format : I*2

Etat-civil-patient

Définition : Valeur qui spécifie l'état civil du patient

Valeurs possibles : 0 = inconnu
1 = célibataire
2 = marié
3 = veuf/veuve
4 = séparé

Format : I*2

Médecin-patient

Définition : Valeur de référence dans la table des médecins

Format : I*2

Nom-médecin-patient

Définition : Nom du médecin de famille du patient

Format : 20A*1

Prénom-médecin-patient

Définition : Prénom du médecin de famille du patient

Format : 6A*1

Organisation-inscription

Définition : Valeur qui définit l'organisation dans laquelle le patient a été inscrit.

Valeurs possibles :

- 1 = clinique Saint-Jean
- 2 = clinique Sainte-Anne
- 3 = laboratoire dr X.

Format : 1A*1

Dernier-numéro-admission-patient

Définition : Valeur du dernier numéro d'admission qui concerne le patient dans l'organisation

Format : I*2

Date-dernière-admission

Définition : Date à laquelle le patient a été hospitalisé pour la dernière fois dans l'organisation.

Format : I*2

Chambre-lit-patient

Définition : Numéro de la chambre et du lit dans lequel se trouve le patient hospitalisé

Format : I*2

Classe-chambre-patient

Définition : Valeur qui définit le type de chambre dans laquelle le patient est hospitalisé

- Valeurs possibles :
- 0 = inconnue
 - 1 = chambre "privée"
 - 2 = chambre à deux lits
 - 3 = "salle"

Format : I*2

Date-dernier-transfert-patient

Définition : Date à laquelle le dernier transfert a été effectué pour le patient lors de sa dernière admission dans l'organisation

Format : I*2

Heure-dernier-transfert-patient

Définition : Heure à laquelle le dernier transfert a été effectué pour le patient lors de sa dernière admission dans l'organisation

Format : I*2

1.3.1.2. Entité DEMANDE

Numéro-patient-dem

Définition : Valeur de référence du patient pour qui la demande a été prescrite et encodée

Format : DI*4

Numéro-dem

Définition : Numéro de la demande d'analyses qui l'identifie parmi toutes les demandes de l'organisation

Format : 2DI*4

Institution-hospitalière-dem

Définition : Si le patient pour qui la demande a été prescrite est hospitalisé, cet attribut désigne la valeur de référence de l'institution hospitalière selon les normes légales.

Format : I*2

Service-hospitalier-dem

Définition : Valeur de référence, du service légal

dans lequel le patient est hospitalisé
au moment de l'encodage de la demande
qui le concerne.

Format : I*2

Organisation-inscription-dem

Définition : Valeur de référence de l'organisation
dans laquelle la demande est encodée.

Format : I*2

Localisation-dem

Définition : Valeur qui détermine la localisation
physique du laboratoire dans lequel est
encodé la demande.

Format : I*2

Situation-patient-dem

Définition : Valeur qui définit la situation du pa-
tient au moment de l'encodage de sa de-
mande. Valeurs possibles :
1 = patient consultant
2 = patient hospitalisé
3 = patient "apporté"

Format : I*1

Statut-dem

Définition : Valeur qui définit le statut de la de-
mande. Valeurs possibles :
bit 01-02 : au moins une analyse
demandée
03-04 : au moins 1 analyse réceptionnée
05-06 : " " éditée sur
liste
07-08 : " " répondue
09-10 : " " validée
11-12 : " " éditée sur
protocole
13-14 : " " ne reçoit
pas res.
15 : " " annulée
16 : " " supprimée

Format : I*2

Numéro-dr1-dem

Définition : Valeur de référence qui identifie le médecin qui a prescrit la demande et pour qui le protocole sera édité.

Format : I*2

Numéro-dr2-dem/Numéro-dr3-dem/Numéro-dr4-dem

Définition : Valeur de référence qui identifie le médecin pour qui une deuxième/troisième/quatrième copie du protocole de la demande doit être envoyée.

Format : I*2

Date-dem

Définition : Date à laquelle la demande a été prescrite.

Format : I*2

Heure-dem

Définition : Heure à laquelle la demande a été prescrite.

Format : I*2

Numéro-user-dem

Définition : Numéro du user qui a encodé et créé la demande.

Format : I*2

Date-première-réception

Définition : Date à laquelle on réceptionne le premier échantillon qui correspond à la demande.

Format : I*2

Heure-première-réception

Définition : Heure à laquelle on réceptionne le premier échantillon qui correspond à la demande.

Format : I*2

Date-validation

Définition : Date de la dernière validation effectuée pour cette demande.

Format : I*2

Date-protocole-dem

Définition : Date de la dernière édition du protocole qui concerne la demande.

Format : I*2

Date-facturation-dem

Définition : Date du calcul de la facture qui concerne la demande.

Format : I*2

Numéro-admission-dem

Définition : Numéro d'admission du patient concerné par la demande au moment de la création de cette demande. Valeurs possibles :
0 = Patient non hospitalisé
<>0 = Numéro admission pat. hospit.

Format : DI*4

Service-dem

Définition : Valeur de référence du service dans lequel le patient se trouve au moment de la création de la demande qui lui est prescrite. Valeurs possible :
0 = Patient non hospitalisé
<>0 = Valeur réf. service

Format : I*2

Numéro-chambre-dem

Définition : Valeur qui désigne le numéro de la chambre et du lit dans lequel se trouve le patient hospitalisé, pour qui la demande a été prescrite.

Valeurs possibles :

0 = Patient non hospitalisé

<>0 : Valeur réf. chambre et lit

Format : I*2

Date-admission-dem

Définition : Date à laquelle le patient a été admis pour la dernière fois, et pour qui la demande a été prescrite.

Valeurs possibles :

0 = Patient non hospitalisé

<>0 = Valeur date dernière admission

Format : I*2

Heure-admission-dem

Définition : Heure de la dernière admission du patient pour qui la demande a été prescrite. Valeurs possibles :

0 = patient non hospitalisé

<>0 = valeur heure dernier admission

Format : I*2

1.3.1.3. Entité ANALYSE

Code Analyse

Définition : Code qui identifie une analyse dans la base de données.

Format : 4A*1

Nombre-résultats-analyse

Définition : Nombre de résultats qui sont attendus pour cette analyse.

Format : I*2

Libellé-fr-analyse

Définition : Libellé francophone de l'analyse

Format : 16A*1

Libellé-nl-analyse

Définition : Libellé néerlandophone de l'analyse

Format : 16A*1

Type-résultat-analyse

Définition : Valeur qui détermine le type de résultat attendu de l'analyse

Valeurs possibles :

- 1 = numérique
- 2 = type dilution
- 4 = "body-site"
- 5 = "P/N/D/A" (positif,...)
- 6 = culture
- 7 = RAST (allergènes)
- 8 = Antibiogramme
- 9 = Expression
- 10 = Texte libre (avec/sans expr.)

Format : I*2

Unité-analyse

Définition : Unité dans laquelle est exprimée le résultat de l'analyse (type numérique).

Format : 10A*1

Val-impos-inf-ana

Définition : Valeur en deça de laquelle un résultat est refusé (type numérique ou dilution)

Format : R*4

Val-impos-sup-ana

Définition : Valeur au-delà de laquelle un résultat est refusé (type numérique ou dilution)

Format : R*4

Body-site-def-ana

Définition : Valeur de défaut, pour une analyse de type "BS", d'un body-site.

Valeurs possibles :

0 = pas de body-site de défaut

<>0 = body-site de défaut

Format : DI*4

Flag-body-site-def-ana

Définition : Valeur qui définit l'acceptation automatique du body-site à la création de la demande. Valeurs possibles :

0 = pas d'acceptation du body-site à la création de la demande

<>0 = acceptation nécessaire du body-site à la création de la demande

Format : I*2

1.3.1.4. Entité BODY-SITE

Code-body-site

Définition : Code qui identifie le site de prélèvement de l'échantillon bactériologique.

Format : 8A*1

Libellé-fr-body-site

Définition : Libellé du site de prélèvement exprimé en français

Format : 24A*1

Libellé-nl-body-site

Définition : Libellé du site de prélèvement exprimé en néerlandais

Format : 24A*1

Libellé-ssf-fr-body-site

Définition : Libellé de la sous-famille exprimé en français, de la sous-famille à laquelle appartient le site de prélèvement.

Format : 16A*1

Libellé-ssf-nl-body-site

Définition : Libellé de la sous-famille exprimé en néerlandais, de la sous-famille à laquelle appartient le site de prélèvement.

Format : 16A*1

Body-site-master

Définition : Valeur de référence du body-site "master" auquel le body-site est associé. Valeurs possibles :
0 = Body-site est "master"
<X> = Body-site "master" auquel est relié le body-site

Format : DI*4

Def-num-body-site

Définition : Valeur de référence qui définit si la numération porte sur toutes les souches du body-site ou si elle porte sur une couche bien précise.
Valeurs possibles :
1 = Type de num. "générale"
2 = Type de num. "par souche"

Format : I*2

Type-num-body-site

Définition : Valeur de référence qui spécifie si la numération qui concerne le body-site est qualitative ou quantitative.
Valeurs possibles :
1 = Type de num. qualitative
2 = Type de num. quantitative

Format : I*2

Num-min-body-site

Définition : Numération qualitative ou quantitative en deçà de laquelle le résultat de la culture sur ce body-site n'est pas accepté.

Format : DI*4

Num-max-body-site

Définition : Numération qualitative ou quantitative au-delà de laquelle le résultat de la culture sur ce body-site n'est pas accepté.

Format : DI*4

1.3.1.5. Entité GERME

Code-germe

Définition : Code qui identifie un germe dans la base de données.

Format : 8A*1

Genre-germe

Définition : Genre auquel appartient le germe dans la B.D.

Format : 16A*1

Espèce-germe

Définition : Espèce à laquelle appartient le germe dans la B.D.

Format : 16A*1

Serotype-germe

Définition : Serotype qui caractérise le germe dans la B.D.

Format : 10A*1

Forme-germe

Définition : Forme du germe vu à l'examen microscopique

Format : 14A*1

Coloration-Gram

Définition : Coloration Gram du germe qui caractérise ses propriétés au niveau de la constitution de sa membrane externe et qui permet une première classification grossière des germes.

Format : 14A*1

Libellé-fr-germe

Définition : Libellé du germe exprimé en français.

Format : 32A*1

Libellé-nl-germe

Définition : Libellé du germe exprimé en néerlandais

Format : 32A*1

Bitst-gen-germe

Définition : Valeur de référence du germe qui spécifie s'il s'agit ou non d'un germe identifié définitivement ou non.

Valeurs possibles :

bit 0 = germe identification définitive

bit 1 = germe "grossier" non définitif

Format : I*2

1.3.1.6. Entité ANTIBIOTIQUE

Code-antibiotique

Définition : Code qui identifie un antibiotique dans l'ensemble des antibiotiques.

Format : 6A*1

Fam-antibiotique

Définition : Code de la famille à laquelle appartient l'antibiotique.

Format : 4A*1

Ssf-antibiotique

Définition : Sous-famille à laquelle appartient l'antibiotique.

Format : 2A*1

Libellé-fr-antibiotique

Définition : Libellé francophone de l'antibiotique.

Format : 40A*1

Libellé-nl-antibiotique

Définition : Libellé néerlandophone de l'antibiotique.

Format : 40A*1

Date-création-update-antibiotique

Définition : Date de création ou de mise à jour des informations concernant l'antibiotique.

Format : I*2

1.3.1.7. Entité ANTIBIOGRAMME

Code-antibiogramme

Définition : Code qui identifie un antibiogramme dans l'ensemble des antibiogrammes.

Format : 6A*1

Nombre-antibiotique

Définition : Nombre d'antibiotiques qui constituent l'antibiogramme.

Format : I*2

Antibiotiques

Définition : Valeur de référence des 14 antibiotiques (valeur maximale) qui constituent l'antibiogramme.

Format : 14I*2

1.3.1.8. Entité JOURNALIERE

Body-site-dem-jrn

Définition : Référence du body-site le plus précis possible, de l'échantillon bactériologique appartenant au résultat de la demande journalière.

Format : DI*4

Germe-dem-jrn

Définition : Référence du germe concerné dans la demande bactériologique "journalière" à traiter.

Format : DI*4

Patient-dem-jrn

Définition : Référence du patient concerné par la demande bactériologique "journalière" à traiter.

Format : DI*4

Gestion-dem-jrn

Définition : Statut de la demande bactériologique "journalière" à traiter.

Format : I*2

Date-write-dem-jrn

Définition : Date de l'écriture de la demande bactériologique "journalière" à traiter.

Format : I*2

Heure-write-dem-jrn

Définition : Heure de l'écriture de la demande bactériologie à traiter.

Format : I*2

Date-adm-dem-jrn

Définition : Date d'admission de la demande bactériologique "journalière" à traiter.

Format : I*2

Date-dem-jrn

Définition : Date de la demande bactériologique journalière à traiter.

Format : I*2

Numéro-adm-dem-jrn

Définition : Numéro d'admission du patient concerné dans la demande journalière à traiter.

Format : DI*4

Numéro-dem-jrn

Définition : Numéro de la demande bactériologique journalière à traiter.

Format : 2DI*4

Situation-dem-jrn

Définition : Situation de la demande bactériologique journalière à traiter.

Valeurs possibles :

0 = demande pour patient ambulant

1 = demande pour patient hospitalisé

Format : I*2

Service-dem-jrn

Définition : Référence du service infirmier où se situe le patient au moment du prélèvement de la demande journalière bactériologique à traiter.

Format : I*2

Numéro-chambre-dem-jrn

Définition : Référence du numéro de la chambre où se situe le patient au moment du prélèvement de la demande bactériologique à traiter.

Format : I*2

Numéro-ana-body-site-dem-jrn

Définition : Numéro de l'analyse qui concerne le site de prélèvement de la demande bactériologique à traiter.

Format : I*2

Numéro-ana-culture-dem-jrn

Définition : Numéro de l'analyse qui concerne la culture de la demande bactériologique à traiter.

Format : I*2

Numéro-ana-abg-dem-jrn

Définition : Numéro de l'analyse qui concerne l'antibiogramme de la demande bactériologique à traiter.

Format : I*2

Non-redondance-dem-jrn

Définition : Valeur de référence qui détermine la non redondance du résultat bactériologique, par comparaison aux résultats bactériologiques déjà répondus pour le même patient.

Valeurs possibles :

0 = résultat redondant

1 = résultat non redondant ("mère")

Format : I*2

1.3.1.9. Entité EPIDEMIOLOGIE

Body-site-épidémio

Définition : Référence du body-site master, qui concerne les demandes bactériologiques d'un même patient, où un même germe à été identifié.

Format : DI*4

Germe-épidémio

Définition : Référence du germe qui concerne les demandes bactériologiques d'un même patient, pour un même site de prélèvement "master".

Format : DI*4

Pat-épidémio

Définition : Référence du patient qui concerne ses demandes bactériologiques.

Format : DI*4

Statut-mère-épidémio

Définition : Référence qui définit le statut de la dernière occurrence du germe, dans un même body-site "master" pour un même patient.

Valeurs possibles :

bit 0 = "mère" éditée pathologique

bit 1 = "mère" éditée cible

Format : I*2

Date-premier-write-dem-epidemio

Définition : Date de la première occurrence de la demande bactériologique où on a rencontré chez un patient, un germe dans un body-site précis.

Format : I*2

Date-dernier-write-dem-epidemio

Définition : Date de la dernière occurrence de la demande bactériologique où on a rencontré chez le même patient, le même germe, dans le même body-site "master".

Format : I*2

Numéro-dernière-dem-mère-épidémio

Définition : Numéro de référence de la dernière demande pour laquelle le résultat bactériologique est non redondant.

Format : 2DI*4

Date-dernière-mère-épidémio

Définition : Date de la dernière occurrence "mère" de la demande bactériologique où on a rencontré chez le même patient, le même germe, dans le même body-site "master".

Format : I*2

Délai-référence-épidémio

Définition : Délai de non redondance à partir duquel la dernière occurrence du résultat bactériologique pour ce germe, ce site de prélèvement, chez ce patient a été considérée.

Format : I*2

1.3.1.10. Entité SERVICE

Code-service

Définition : Valeur de référence qui définit le service infirmier de la clinique.

Format : 4A*1

Libellé-fr-service

Définition : Libellé du service infirmier exprimé en français.

Format : 30A*1

Libellé-nl-service

Définition : Libellé du service infirmier exprimé en néerlandais.

Format : 30A*1

1.3.2. Les associations

1.3.2.1. Association GENERALITES-BODY-SITE

1.1. Définition :

Une occurrence de cette association décrit les caractéristiques générales du body-site quel que soit le germe qui lui est associé.

1.2. Entités

T.E GERME

T.E BODY-SITE

T.E ANTIBIOGRAMME

1.3. Connectivité

1 - 1 pour GERME

Le germe considéré dans toute occurrence de cette association est le germe nul.

0 - * pour BODY-SITE

0 - * pour ANTIBIOGRAMME

1.4. Attributs

1.4.1. Body-site généralité :

Définition : Valeur de référence du body-site pour lequel des informations tous germes confondus doivent être spécifiées.

Format : DI*4

1.4.2. Germe généralité :

Définition : Valeur de référence du germe qui signifie sémantiquement tous germes confondus. Valeur de référence = 0.

Format : DI*4

1.4.3. Information numération :

Définition : Valeur qui définit le type de numération pour un body-site tous germes confondus. Valeurs possibles :
1 = Numération qualitative
2 = Numération quantitative

Format : I*2

1.4.4. Durée non redondance :

Définition : Durée exprimée en jours qui détermine à partir de quel intervalle de date on peut considérer le germe rencontré dans ce body-site comme non redondant. Cette valeur sert de valeur de défaut pour tous les germes dans ce body-site.

Format : I*2

1.4.5. Numération généralités :

Définition : Numération minimum et maximum (qualitative ou quantitative) relevante pour ce body-site, tous germes confondus.

Format : 2(DI*4)

1.4.6. Date dernier update :

Définition : Date de la dernière mise à jour des informations concernant ce germe dans ce body-site.

Format : I*2

1.4.7. User dernier update :

Définition : Numéro du dernier utilisateur qui a porté les dernières informations concernant ce body-site tous germes confondus.

Format : I*2

1.4.8. Antibiogramme :

Définition : Valeur de référence de(s) antibiogrammes généralement testés lors des dosages des analyses bactériologiques dans ce body-site.

Format : x(I*2)

1.3.2.2. Association PARTICULARITE-GERME - BODY-SITE

1.1. Définition :

Une occurrence de cette association décrit les caractéristiques générales du germe rencontré dans ce body-site.

1.2. Entités

T.E GERME

T.E BODY-SITE

1.3. Connectivité

- 0 - * pour GERME Un germe peut exister sans être associé spécifiquement à un body-site
- 1 - * pour BODY-SITE site particulier.

1.4. Attributs

1.4.1. Body-site particularité

Définition : Valeur de référence du body-site pour lequel des informations pour un germe particulier doivent être spécifiées.

Format : DI*4

1.4.2. Germe particulier

Définition : Valeur de référence du germe particulier pour lequel on spécifie des informations qui le concernent dans un certain body-site.

Format : DI*4

1.4.3. Information numération

Définition : Valeur qui définit le type de numération pour un germe particulier dans un body-site. Valeurs possible :

- 1 = Numération qualitative
- 2 = Numération quantitative

Format : I*2

1.4.4. Durée non redondance

Définition : Durée exprimée en jours qui détermine à partir de quelle date on peut considérer le germe comme non redondant.

Format : I*2

1.4.5. Numération généralité germe

Définition : Numération minimum et maximum (qualitative ou quantitative) de ce germe particulier dans un body-site précis. Toute numération inférieure ou supérieure à ces valeurs est considérée comme une erreur.

Format : 2(DI*4)

1.4.6. Date dernier update

Définition : Date de la dernière mise à jour des informations concernant ce germe dans ce body-site.

Format : I*2

1.4.7. User dernier update

Définition : Numéro du dernier utilisateur qui a porté les dernières informations concernant ce germe dans ce body-site.

Format : I*2

1.3.2.3. Association GERME-PATHOGENE

1.1. Définition :

Une occurrence de cette association désigne un germe dans un body-site comme étant pathogène.

1.2. Entités

T.E GERME

T.E BODY-SITE

T.E ANTIBIOTIQUE

1.3. Connectivité

- 0 - * pour GERME Un germe peut exister sans être pathogène
0 - * pour BODY-SITE Un body-site "slave" ne peut participer à l'occurrence d'une association "germe pathogène"
0 - * pour ANTIBIOTIQUE

1.4. Attributs

1.4.1. Body-site pathogène

Définition : Valeur de référence d'un body-site pour lequel des informations concernant la pathogénicité d'un germe doivent être spécifiées. Valeurs possibles :
0 = tous les body-sites
<>0 = un body-site particulier "master"

Format : DI*4

1.4.2. Germe pathogène

Définition : Valeur de référence du germe particulier dans un body-site pour lequel on spécifie des informations qui déterminent la pathogénicité de celui-ci dans ce dernier.

Format : DI*4

1.4.3. Délai-édition-infection-pathogène

Définition : Délai après lequel le redondant d'une "mère" pathogène est à nouveau édité.

Format : I*2

1.4.4. Numération-pathogène

Définition : Numération minimum et maximum (qualitative ou quantitative) à partir de laquelle on considère le germe comme pathogène dans ce body-site.

Format : 2 (DI*4)

1.4.5. Date-dernier-update-pathogène

Définition : Date de la dernière mise à jour des informations concernant ce germe dans ce body-site, considéré comme pathogène dans certaines conditions.

Format : I*2

1.4.6. User-dernier-update-pathogène

Définition : Numéro du dernier utilisateur qui a porté les dernières informations concernant ce germe dans ce body-site, considéré comme pathogène dans certaines conditions.

Format : I*2

1.4.7. Antibiotique-pathogène

Définition : Valeur de référence d'un antibiotique qui est un critère significatif pour désigner ce germe comme étant pathogène.

Format : x(I*2)

1.4.8. Résultat-antibiotique-pathogène

Définition : Résultat de la réaction du germe à un certain antibiotique considéré comme significatif pour désigner ce germe comme pathogène.

Format : x(I*2)

1.3.2.4. Association GERME-CIBLE

1.1. Définition :

Une occurrence de cette association désigne un germe dans un body-site comme étant cible, c'est-à-dire, un germe susceptible d'être transmis d'un patient à un autre.

1.2. Entités

T.E GERME

T.E BODY-SITE

T.E ANTIBIOTIQUE

1.3. Connectivité

- 0 - * pour GERME Un germe peut exister sans être cible.
- 0 - * pour BODY-SITE Un body-site "slave" ne peut participer à l'occurrence d'une association "germe cible".
- 0 - * pour ANTIBIOTIQUE

1.4. Attributs

1.4.1. Body-site-cible

Définition : Valeur de référence du body-site pour lequel des informations concernant l'aspect cible d'un germe doivent être spécifiées. Valeurs possibles :
0 = tous les body-sites
<>0 = un body-site particulier "master"

Format : DI*4

1.4.2. Germe-cible

Définition : Valeur de référence du germe particulier dans un body-site pour lequel on spécifie des informations qui déterminent l'aspect cible de ce germe dans ce body-site.

Format : DI*4

1.4.3. Durée-recherche-infection-croisée

Définition : Durée exprimée en jours qui détermine sur quelle période la recherche des candidats de "cross-infection" doit être effectuée pour ce germe considéré comme cible.

Format : I*2

1.4.4. Numération-cible

Définition : Numération minimum et maximum (qualitative ou quantitative) à partir desquelles le germe peut être considéré comme cible dans ce body-site.

Format : 2(DI*4)

1.4.5. Date-dernier-update-pathogène

Définition : Date de la dernière mise à jour des informations concernant ce germe dans ce body-site, considéré comme cible dans certaines conditions.

Format : I*2

1.4.6. User-dernier-update-pathogène

Définition : Numéro du dernier utilisateur qui a porté les dernières informations concernant ce germe dans ce body-site, dans certaines conditions.

Format : I*2

1.4.7. Antibiotique-cible

Définition : Valeur de référence d'un antibiotique qui est un critère significatif pour désigner ce germe comme étant cible.

Format : x(I*2)

1.4.8. Résultat-antibiotique-pathogène

Définition : Résultat de la réaction du germe à un certain antibiotique considéré comme significatif pour désigner ce germe comme cible.

Format : x(I*2)

1.3.2.5. Association "RESULTAT-ANALYSE"

1.1. Définition :

Un occurrence de cette association signifie qu'une analyse dans la demande a été répondue et possède un résultat.

Remarque : La définition même du résultat est différente en fonction du type de résultat de l'analyse. Nous décrirons l'association qui relie la demande à une analyse de type :

4 : type analyse origine de prélèvement

6 : type analyse bactériologie (germe/numération)

8 : type analyse antibiogramme

Par contre, les connectivités aux entités seront détaillées globalement, indépendamment du type de résultat.

1.2. Entités

T.E DEMANDE

T.E ANALYSE

T.E ANTIBIOGRAMME

T.E ANTIBIOTIQUES

T.E BODY-SITES

T.E GERMES

T.E JOURNALIER

1.3. Connectivité

1 - * pour DEMANDE

0 - * pour ANALYSE

0 - * pour GERME

1 - 1 pour BODY-SITE

0 - * pour ANTIBIOGRAMME

0 - * pour ANTIBIOTIQUE

0 - * pour ALLERGENES

1.4. Attributs

a) Attributs de l'association qui concernent une analyse type 4 dans la demande.

a) 1.4.4. Numéro-dem

Définition : Numéro du record de la demande concernée par le résultat de l'analyse.

Format : I*2

a) 1.4.2. Numéro-analyse-dem

Définition : Numéro du record de l'analyse pour qui le résultat a été répondu.

Format : I*2

a) 1.4.3. Statut-analyse-dem

Définition : Statut qui définit le statut de l'analyse dans la demande.

Format : I*2

a) 1.4.4. Position-analyse-station-dem

Définition : Position de l'analyse dans la station, pour qui un résultat a été répondu.

Format : I*2

a) 1.4.5. Position-analyse-secteur-dem

Définition : Position de l'analyse dans le secteur de validation pour laquelle un résultat a été répondu.

Format : I*2

a) 1.4.6. Body-site-analyse-dem

Définition : Valeur de référence du body-site de l'analyse bactériologique dans la demande.

Format : DI*4

b) Attributs de l'association qui concerne une analyse type 6 dans la demande.

b) 1.4.1. Numéro-dem

(idem a) 1.4.1.)

b) 1.4.2. Numéro-analyse-dem

(idem a) 1.4.2.)

b) 1.4.3. Statut-analyse-dem

(idem a) 1.4.3.)

b) 1.4.4. Position analyse dans la station

(idem a) 1.4.4.)

b) 1.4.5. Position analyse dans le secteur

(idem a) 1.4.5.)

b) 1.4.6. Résultat-global-analyse

Définition : Valeur qui désigne le résultat de l'analyse. Valeur possible :

- 0 = valeur initiale
- 1 = valeur négative
- 2 = valeur négative à ne pas éditer sur protocole
- 3 = absence de germes
- 5 = contamination
- 1 = positif
- 2 = positif à ne pas éditer sur protocole

Format : I*2

b) 1.4.7. Bitstring gestion divers

Définition : Valeur de référence qui exprime diverses informations concernant les différents types de réponses possibles annexés à l'analyse.

Format : I*2

b) 1.4.8. Numération générale

Définition : Résultat de la numération, laquelle est générale pour le body-site considéré.

Format : 2 DI*4

b) 1.4.9. Germe réponse "générale"

Définition : Valeur de référence du germe qui complète l'information concernant l'analyse bactériologique dont la numération est générale.

Format : DI*4

b) 1.4.9. Date de positivité/négativité analyses

Définition : Valeur qui désigne la date à laquelle le résultat de l'analyse bactériologique a été encodé.

Format : 2(I*2)

b) 1.4.0. Résultat analyse (numération non générale)

Définition : Informations concernant le résultat de l'analyse proprement dite. Ce résultat définit pour chaque germe, la numération lue pour ce germe, le type de numération qui concerne le body-site de cette analyse.

Format : (I*2 : Numéro de la souche
DI*4 : Numération inférieure
DI*4 : Numération supérieure
DI*4 : Valeur de référence du germe) * nombre de souches

c) Attributs de l'association qui concerne une analyse type 8 dans la demande.

c) 1.4.1. Numéro-dem

(idem a) 1.4.1.)

c) 1.4.2. Numéro-analyse-antibiogramme-dem

(idem a) 1.4.2.)

c) 1.4.3. Statut-analyse-dem

(idem a) 1.4.3.)

c) 1.4.4. Position-analyse-station-dem

(idem a) 1.4.4.)

c) 1.4.5. Position-analyse-secteur-dem

(idem a) 1.4.5.)

c) 1.4.6. Antibiotique testé

Définition : Valeur de référence de l'antibiotique
testé pour l'analyse

Format : x(I*2)

c) 1.4.7. Résultat antibiotique testé

Définition : Résultat de l'analyse pour le germe
et l'antibiotique dans la demande.

Valeurs possibles : 1 = sensible
2 = intermédiaire
3 = résistant

Format : I*2

1.3.2.6. Association JOURNALIER-EPIDEMIO

1.1. Définition :

Une occurrence de cette association permet de relier
les résultats bactériologiques du jour, aux résultats
bactériologiques qui ont été traités précédemment.

1.2. Entités

T.E DEMANDE-JOURNALIERE

T.E EPIDEMIO

1.3. Connectivité

1 - * pour DEMANDE-JOURNALIERE

0 - * pour EPIDEMIO

1.4. Attributs

1.4.1. Body-site-journalier

Définition : Valeur de référence du body-site dans la
demande bactériologique

Format : DI*4

1.4.2. Germe-journalier

Définition : Valeur de référence du germe dans la demande bactériologique.

Valeurs possibles

0 = culture négative

<>0 = référence au germe définitif

Format : DI*4

1.4.3. Patient-journalier

Définition : Valeur de référence du patient concerné dans la demande bactériologique journalière.

Format : DI*4

1.4. Contraintes d'intégrité

1.4.1. Contraintes d'existence

La participation d'une occurrence d'un T.E GERME dans une occurrence du type d'association JOURNALIER-EPIDEMIO n'existe que si ce germe est définitif ou nul.

La participation d'une occurrence d'un GERME dans une occurrence du type d'association GERME-CIBLE n'existe que si ce germe est définitif ou nul.

La participation d'une occurrence d'un GERME dans une occurrence du type d'association GERME-PATHOGENE/PARTICULARITE-GERME-BODY-SITE GENERALITE-BODY-SITE n'existe que si ce germe est définitif ou nul.

Une occurrence du type d'association GERME-CIBLE/GERME-PATHOGENE/PARTICULARITE-GERME-BODY-SITE/ n'existe que si le body-site associé à cette occurrence est un body-site "master".

1.4.2. Contraintes d'inclusion

La participation d'une occurrence du type d'entité BODY-SITE dans une occurrence du type d'association RESULTAT-ANALYSE inclut sa participation dans une occurrence du type d'association JOURNALIER-EPIDEMIO à condition que cette analyse n'aie pas été annulée ou supprimée et que le germe associé est définitif ou nul.

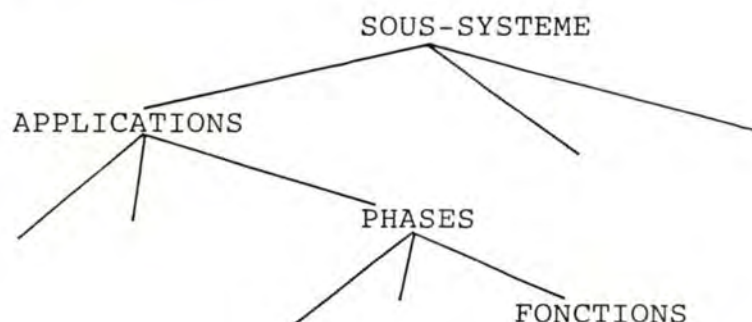
La participation d'une occurrence du type d'entité BODY-SITE dans une occurrence du type d'association GERME-CIBLE n'exclut pas sa participation à une occurrence du type d'association GERME-PATHOGENE/ ou PARTICULARITES-GERME-BODY-SITE ou GENERALITES-BODY-SITE.

Il en va de même pour le T.E GERME.

2. DESCRIPTION DES TRAITEMENTS

2.1. Méthodologie

La méthode d'analyse appliquée fournit une hiérarchie dans la spécification des traitements.



- Le sous-système constitue le cadre général du projet.
- Une application est un domaine du projet qui est indépendant des autres.
- Une phase est une unité spatio-temporelle.
- Une fonction est une action élémentaire.

2.2. Décomposition du projet en applications

Pour le projet d'épidémiologie, nous obtenons pour le sous-système PROJET EPIDEMIOLOGIE, deux applications :

- Application BACTERIOLOGIE
- Application CONTROLE DES INFECTIONS

2.3. Décomposition des applications en phases

2.3.1. Description et spécifications des phases

Les phases seront dégagées à partir du diagramme des flux de la solution projetée et feront l'objet d'une description en termes de :

- objectif
- performance
- consultation du S.I
- action sur la mémoire du S.I

Objectif

L'objectif reprend en quelques mots le rôle que joue la phase dans l'application.

Performance

La performance décrit brièvement les caractéristiques du traitement telles que la fiabilité des informations, le temps de réponse, la lisibilité et la "vérifiabilité" de l'information, la sécurité du traitement de l'information.

Consultation du S.I

Elle spécifie les informations du système d'information que la phase consulte pour réaliser son objectif.

Action sur la mémoire du S.I

Il s'agit de spécifier les informations du S.I que la phase manipule et met à jour pour réaliser son objectif.

Nous avons pris l'option de ne pas décrire toutes les phases qui existent dans l'application bactériologie. En effet, parmi ces phases, certaines sont nouvelles, d'autres par contre ont fait l'objet d'adaptation et ont subi aucun changement quant à la structuration et la sémantique de leurs informations.

Nous décrirons plus précisément les phases qui sont liées directement à l'application CONTROLE DES INFECTIONS.

2.3.2. Les différentes phases

Phases de l'application BACTERIO

- * - INTRODUCTION DES RESULTATS
- VALIDATION DES RESULTATS
- EDITION DES RESULTATS
- ACCEPTATION COMMANDES
- ** - CREATION / MISE-A-JOUR DES GERMES
- ** - EDITION DES GERMES
- ** - CREATION / MISE-A-JOUR DES BODY-SITES

** - EDITION DES BODY-SITES

** - CREATION / MISE-A-JOUR DES BODY-SITES MASTER/SLAVE

Phases de l'application CONTROLE DES INFECTIONS

** - CREATION /MAJ GENERALITES BODY-SITE GERMES CONFONDUS

** - EDITION GENERALITES BODY-SITE GERMES CONFONDUS

** - CREATION /MAJ GERME PARTICULIER DANS BODY-SITE

** - EDITION GERME PARTICULIER DANS BODY-SITE

** - CREATION /MAJ DES GERMES CIBLES

** - EDITION DES GERMES CIBLES

** - CREATION /MAJ DES GERMES PATHOGENES

** - EDITION DES GERMES PATHOGENES

** - ACCEPTATION COMMANDES EPIDEMIOLOGIE

** - EDITIONS LISTES INFECTIONS TRIEES

** - EDITION / AFFICHAGE STATISTIQUES

** - TRAITEMENT JOURNALIER

Les phases marquées ** seront décrites et décomposées en fonctions. La phase marquée * sera simplement décrite.

2.3.3. Description des phases spécifiées

PHASE 1 : INTRODUCTION DES RESULTATS

A) Objectif :

A tout moment, le médecin comme le technicien peut enregistrer les nouveaux résultats (culture, identification des germes, les numérations, et les résultats antibiogrammes).

Si la validation est correcte, le système mémorise le(s) résultat(s) de bactériologie.

B) Performance :

Enregistrer le plus rapidement possible le résultat et assurer une bonne validation de ceux-ci.

C) Consultation du S.I :

T.E GERME
T.E ANALYSE
T.E ANTIBIOGRAMME
T.E ANTIBIOTIQUE
T.E BODY-SITE

D) Action sur la mémoire du S.I :

T.A RESULTAT-ANALYSE
T.E JOURNALIER

E) Ecran :

Voir annexe

PHASE 2 : ACCEPTATION COMMANDES

A) Objectif :

Proposer à l'utilisateur l'ensemble des commandes possibles et accessibles au personnel du laboratoire.

B) Performance :

Clarté des informations attendues et intégrité des commandes.

C) Consultation du S.I :

Néant

D) Action sur la mémoire du S.I :

Néant

PHASE 3 : CREATION /MAJ GERME

A) Objectif :

A tout moment, le médecin comme le technicien peut enregistrer un nouveau germe. Il introduit le code du germe, et le système vérifie :

- si le germe existe déjà dans la base de données
- si les conditions de validité sont remplies.

Si la validation est correcte, le système mémorise les informations qui concernent ce germe.

B) Performance :

Enregistrer rapidement et correctement les informations du germe.

C) Consultation du S.I :

T.E GERME

D) Action sur la mémoire du S.I :

T.E GERME

E) Ecran :

Voir annexe

PHASE 4 : EDITION GERME

A) Objectif :

Editer sur le terminal choisi par l'utilisateur, ou afficher à son écran les informations qui concernent le(s) germe(s) qu'il a précisé par le code.

B) Performance :

Exactitude et clareté dans l'édition des informations.

C) Consultation du S.I :

T.E GERME

D) Action sur la mémoire du S.I :

Néant

E) Liste :

Voir annexe

PHASE 5 : CREATION /MAJ BODY-SITE

A) Objectif :

A tout moment, le médecin comme le technicien peut enregistrer un nouveau site de prélèvement. Il introduit le code du site de prélèvement, et le système vérifie :
- si le site existe déjà dans la base de données
- si les conditions de validité sont remplies.

Si la validation est correcte, le système mémorise les informations qui concernent ce site de prélèvement.

B) Performance :

Enregistrer rapidement et correctement les renseignements des sites de prélèvements.

C) Consultation du S.I :

T.E BODY-SITE

D) Action sur la mémoire du S.I :

T.E BODY-SITE

E) Ecran :

Voir annexe

PHASE 6 : EDITION BODY-SITE

A) Objectif :

Editer sur le terminal choisi par l'utilisateur, ou afficher à son écran la liste des informations qui caractérisent le(s) site(s) de prélèvement désigné par leur code.

B) Performance :

Exactitude et clareté dans l'édition des informations.

C) Consultation du S.I :

T.E BODY-SITE

D) Action sur la mémoire du S.I :

Néant

E) Liste :

Voir annexe

PHASE 7 : CREATION/MISE-A-JOUR BODY-SITE MASTER/SLAVE

A) Objectif :

Créer la relation sémantique d'un body-site considéré "slave" par rapport à un autre body-site "master".

B) Performance :

Enregistrer correctement et rapidement les informations.

C) Consultation du S.I :

T.E BODY-SITE

D) Action sur la mémoire du S.I :

T.E BODY-SITE

E) Ecran :

Voir annexe

PHASE 8 : CREATION /MAJ GENERALITES BODY-SITE

A) Objectif :

Créer ou modifier les informations générales qui concernent un body-site, indépendamment des germes. Ces informations déterminent le type de numération dans ce body-site, les valeurs relevantes de cette numération, les codes d'antibiogrammes susceptibles d'être testé pour ce body-site.

B) Performance :

Enregistrer et valider rapidement les informations qui concernent le body-site.

C) Consultation du S.I :

T.E GERME
T.E BODY-SITE
T.E ANTIBIOGRAMMES
T.A GENERALITES-BODY-SITE

D) Action sur la mémoire du S.I :

T.A GENERALITES-BODY-SITE

E) Ecran :

Voir annexe

PHASE 9 : EDITION GENERALITES BODY-SITE

A) Objectif :

Editer sur le terminal choisi par l'utilisateur, ou afficher à son écran la liste des informations générales qui concernent tous les body-sites, ou un body-site précisé par son code.

B) Performance :

Exactitude et clareté dans l'édition des informations.

C) Consultation du S.I :

T.E BODY-SITE
T.A GENERALITE BODY-SITE

D) Action sur la mémoire du S.I :

Néant

E) Liste :

Voir annexe

PHASE 10 : CREATION /MAJ PARTICULARITE GERME BODY-SITE

A) Objectif :

A tout moment, le médecin comme le technicien peut enregistrer des informations générales concernant un germe bien précis dans certain body-site. Il introduit le code du site de prélèvement, le code du germe et le système vérifie si :

- si le site de prélèvement existe déjà dans la base de données
- si le germe existe déjà dans la base de données
- si les conditions de validité sont remplies.

Si la validation est correcte, le système mémorise les informations qui concernent ce germe dans ce body-site.

B) Performance :

Enregistrer rapidement les informations qui concernent le germe dans ce body-site.

C) Consultation du S.I :

T.E GERME
T.E BODY-SITE
T.A PARTICULARITE-GERME-BODY-SITE

D) Action sur la mémoire S.I :

T.A PARTICULARITE-GERME-BODY-SITE

E) Ecran :

Voir annexe

PHASE 11 : EDITION PARTICULARITE-GERME-BODY-SITE

A) Objectif :

Editer sur le terminal choisi par l'utilisateur, ou afficher à son écran les informations qui concernent le(s) germe(s) particuliers dans tous les body-site ou celui qui a été précisé par son code.

B) Performances :

Exactitude et clareté dans l'édition des informations.

C) Consultation du S.I :

T.E GERME
T.E BODY-SITE
T.A PARTICULARITE-GERME-BODY-SITE

D) Action sur la mémoire du S.I :

Néant

E) Liste :

Voir annexe

PHASE 12 : CREATION /MAJ GERME-CIBLE

A) Objectif :

A tout moment, le médecin comme le technicien peut enregistrer un nouveau germe cible, c'est-à-dire un germe susceptible d'être l'agent d'une infection croisée entre patients hospitalisés. L'utilisateur introduit le body-site et le germe associé, le système vérifie :

- si le site de prélèvement existe déjà dans la base de données
- si le germe existe déjà dans la base de données
- si les conditions de validité sont remplies.

Si la validation est correcte, le système mémorise les informations qui concernent ce germe cible.

B) Performance :

Enregistrer rapidement les germes hospitaliers les plus significatifs.

C) Consultation du S.I. :

T.E BODY-SITE
T.E ANTIBIOTIQUES
T.E GERME
T.A GERME-CIBLE

D) Action sur la mémoire du S.I :

T.A GERME-CIBLE

E) Ecran :

Voir annexe

PHASE 13 : EDITION GERME-CIBLE

A) Objectif :

Editer sur le terminal choisi par l'utilisateur, ou afficher à son écran la liste des germes cibles et les informations qui concernent ce(s) germe(s) cible(s) précisé(s).

B) Performance :

Exactitude et clareté dans l'édition des informations.

C) Consultation du S.I :

T.A GERME-CIBLE
T.E ANTIBIOTIQUE
T.E GERME
T.E BODY-SITE

D) Action sur la mémoire sur S.I :

Néant

E) Liste :

Voir annexe

PHASE 14 : CREATION /MAJ GERME PATHOGENE

A) Objectif :

A tout moment, le médecin comme le technicien peut enregistrer un nouveau germe pathogène. L'utilisateur introduit le body-site et le germe associé, le système vérifie :

- si le site de prélèvement existe déjà dans la base de données
- si le germe existe déjà dans la base de données
- si les conditions de validité sont remplies.

Si la validation est correcte, le système mémorise les informations qui concernent ce germe considéré comme pathogène.

B) Performance :

Enregistrer rapidement les germes pathogènes.

C) Consultation du S.I :

T.E GERME
T.E BODY-SITE
T.E ANTIBIOTIQUE
T.A GERME-PATHOGENE

D) Action sur la mémoire du S.I :

T.A GERME-PATHOGENE

E) Ecran :

Voir annexe

PHASE 15 : EDITION GERME PATHOGENE

A) Objectif :

Editer sur le terminal choisi par l'utilisateur, ou afficher à son écran la liste des germes pathogènes où le germe qu'il a précisé par son code.

B) Performance :

Exactitude et clareté dans l'édition des informations.

C) Consultation du S.I :

T.E GERME
T.E BODY-SITE
T.E ANTIBIOTIQUE
T.A GERME-PATHOGENE

D) Action sur la mémoire du S.I :

Néant

E) Liste :

Voir annexe

PHASE 16 : ACCEPTATION COMMANDES EPIDEMIOLOGIE

A) Objectif :

Proposer au service informatique l'ensemble des commandes possibles pour l'édition des listes dans le cadre de l'application CONTROLE DES INFECTIONS.

B) Performance :

Facilité d'utilisation des commandes et intégrité de celles-ci.

C) Consultation du S.I :

Néant

D) Action sur la mémoire du S.I :

Néant

E) Liste :

Voir annexe

PHASE 17 : EDITION LISTES TRIEES

A) Objectif :

Editer sur le terminal désigné par l'utilisateur, les listes nominatives des cas d'infection, à partir des données épidémiologiques sélectionnées.

Le système affiche à l'écran les types de listes triées :

- Liste des infections triées sur le germe et le service
- Liste des infections triées sur le service et le germe

L'utilisateur précise la période sur laquelle il désire traiter les informations et spécifie le(s) germe(s), le(s) service(s) qui l'intéressent.

B) Performances :

Editer clairement et rapidement sur le terminal désigné, ou sur écran, la liste des patients qui ont été infectés par le(s) germe(s) désigné(s).

C) Consultation du S.I :

T.E GERME
T.E SERVICE
T.E EPIDEMIOLOGIE

D) Action sur la mémoire du S.I :

Néant

E) Liste :

Voir annexe

PHASE 18 : EDITION / AFFICHAGE STATISTIQUES

A) Objectif :

Les statistiques se font sur l'ensemble des données bactériologiques. Le système affiche à l'écran la liste des statistiques possibles qui sont :

- Liste de répartition des germes par service
- Taux d'infection par service
- Taux d'infection par body-site et par service
- Histogramme de sensibilité des germes
- Polyrésistance des germes aux antibiotiques

B) Performances :

Editer clairement et rapidement sur le terminal désigné, ou sur écran, le résultat des statistiques demandées.

C) Consultation du S.I :

T.E GERME
T.E EPIDEMIOLOGIE
T.E ANTIBIOTIQUE
T.E ANTIBIOGRAMME
T.E BODY-SITE
T.E SERVICE
T.E PATIENT
T.A RESULTAT-ANALYSE

D) Action sur la mémoire du S.I :

Néant

E) Liste :

Voir annexe

PHASE 19 : TRAITEMENT JOURNALIER

A) Obejectif :

Chaque soir, le système traite les résultats bactériologiques du jour, met à jour les résultats d'épidémiologie, et édite les listes journalières qui sont :

- des infections définies comme pathologiques
- des infections susceptibles d'être croisées

B) Performances :

Editer un minimum de cas sur les deux listes, de manière à ne faire apparaître que les cas intéressants et non redondants.

C) Consultation du S.I :

T.E DEMANDES
T.E PATIENT
T.E EPIDEMIOLOGIE
T.E GERME
T.E BODY-SITE
T.E ANTIBIOTIQUE
T.A RESULTAT-ANALYSE
T.A GERME-PATHOGENE
T.A GERME-CIBLE
T.A GENERALITES-BODY-SITE
T.A PARTICULARITE-GERME-BODY-SITE

D) Action sur la mémoire du S.I :

T.E JOURNALIER
T.E EPIDEMIOLOGIE

E) Liste :

Voir annexe

2.4. Décomposition des phases en fonctions

2.4.1. Description et spécification des fonctions

Chaque phase se découpe en fonctions. Celles-ci sont décrites en termes de :

- Entrées
- Traitement
- Sorties

Ne seront décrites que les fonctions qui nous intéressent directement dans le cadre de ce mémoire.

Entrées

Il s'agit de décrire les données que la fonction reçoit pour s'exécuter. On classe en deux groupes ces données :

- messages EXTERNE de type input qui sont fournis par l'utilisateur
- messages INTERNE qui sont fournis par l'application elle-même.

Traitement

Il explique comment la fonction va remplir son rôle, sous forme d'un algorithme de haut niveau.

Sorties

Il s'agit ici, des données que la fonction génère. Il peut s'agir de message EXTERNE de type output qui sont fournis à l'utilisateur ou de message INTERNE.

2.4.2. Description des fonctions

PHASE 2 : ACCEPTATION COMMANDES

FONCTION SAISIE-CDE

Entrée : TYPE-CDE (MESSAGE EXTERNE)

Traitement : Saisie TYPE-CDE
Validation TYPE-CDE
Si TYPE-CDE 'SOS', 'HELP', 'KB', 'TERM',
'FRAN', 'NEDE', 'MUT', 'LIST',
'STOP', 'BYE'
alors message d'erreur
sinon case 'SOS' : TRT-SOS-OK
'HELP' : TRT-HELP-OK
'KB' : TRT-KB-OK
'TERM' : TRT-TERM-OK
'FRAN' : TRT-FRAN-OK
'NEDE' : TRT-NEDE-OK
'MUT' : TRT-MUT-OK
'LIST' : TRT-LIST-OK
'STOP' : TRT-STOP-OK
'BYE' : TRT-BYE-OK

Sorties : Soit message d'erreur (MESSAGE EXTERNE)
Soit TRT-SOS-OK (MESSAGE INTERNE)
TRT-HELP-OK (MESSAGE INTERNE)
TRT-KB-OK (MESSAGE INTERNE)
TRT-TERM-OK (MESSAGE INTERNE)
TRT-FRAN-OK (MESSAGE INTERNE)
TRT-NEDE-OK (MESSAGE INTERNE)
TRT-MUT-OK (MESSAGE INTERNE)
TRT-LIST-OK (MESSAGE INTERNE)
TRT-STOP-OK (MESSAGE INTERNE)
TRT-BYE-OK (MESSAGE INTERNE)

FONCTION SOS

Entrée : TRT-SOS-OK (MESSAGE INTERNE)

Traitement : Afficher à l'écran les commandes possibles associées à la phase.

Sorties : Echo des commandes possibles à l'écran.

FONCTION HELP

Entrée : TRT-HELP-OK (MESSAGE INTERNE)

Traitement : Lire et afficher à l'écran le mode d'emploi qui détaille dans la langue de l'utilisateur, la manière de procéder pour utiliser les fonctions de la phase.

Sorties : Liste du mode d'emploi à l'écran.

FONCTION KB

Entrée : TRT-KB-OK (MESSAGE INTERNE)

Traitement : Sélection de l'écran sur lequel sera édité le résultat de la fonction demandée ultérieurement.

Sorties : -

FONCTION TERM

Entrée : TRT-TERM-OK (MESSAGE INTERNE)

Traitement : Pose la question à l'utilisateur, sur quelle imprimante envoie le résultat de la fonction demandée ultérieurement.

Sorties : Edition sur le terminal choisi.

FONCTION FRAN

Entrée : TRT-FRAN-OK (MESSAGE INTERNE)
LANGUE-USER (MESSAGE EXTERNE)

Traitement : Lire la langue du user
Si langue user = 'néerlandais'
alors langue-user = 'français'
sinon -

Sorties : -

FONCTION NEDE

Entrée : TRT-NEDE-OK (MESSAGE INTERNE)
LANGUE-USER (MESSAGE EXTERNE)

Traitement : Lire la langue du user
Si langue user = 'français'
alors langue-user = 'néerlandais'
sinon -

Sorties : -

FONCTION STOP

Entrée : TRT-STOP-OK (MESSAGE INTERNE)

Traitement : Quitter la phase en cours.

Sorties : -

FONCTION BYE

Entrée : TRT-BYE-OK (MESSAGE INTERNE)

Traitement : Quitte le processeur.

Sorties : -

FONCTION SAISIE-RUBRIQUE

Entrée : NUMERO-RUBRIQUE (MESSAGE EXTERNE)
 TRT-MUT-OK/TRT-LIST-OK (MESSAGE INTERNE)

Traitement : Saisie NUMERO-RUBRIQUE
 Validation NUMERO-RUBRIQUE
 Si NUMERO-RUBRIQUE 1 ou 6
 alors imprimer message erreur
 sinon case NUMERO-RUBRIQUE = 1 : TRT-ABG-OK
 " = 2 : TRT-ABT-OK
 " = 3 : TRT-RAST-OK
 " = 4 : TRT-ALL-OK
 " = 5 : TRT-BODY-OK
 " = 6 : TRT-GERME-OK

Sorties : Soit message d'erreur (MESSAGE EXTERNE)
 Soit

TRT-ABG-OK/TRT-ABG-LIST-OK (MESSAGE INTERNE)
 TRT-ABT-OK/TRT-ABT-LIST-OK (MESSAGE INTERNE)
 TRT-RAST-OK/TRT-RAST-LIST-OK (MESSAGE INTERNE)
 TRT-ALL-OK/TRT-ALL-LIST-OK (MESSAGE INTERNE)
 TRT-BODY-OK/TRT-BODY-LIST-OK (MESSAGE INTERNE)
 TRT-GERME-OK/TRT-GERME-LIST-OK (MESSAGE INTERNE)

Ecran : Voir annexe

FONCTION SEL-BODY

Entrée :

TRT-BODY-OK/TRT-BODY-LIST-OK (MESSAGE INTERNE)
 NUMERO RUBRIQUE (MESSAGE EXTERNE)

Traitement : Saisie NUMERO RUBRIQUE
 Si NUMERO RUBRIQUE 0 ou 6
 alors message erreur
 sinon

CASE NUMERO RUBRIQUE = 1 : TRT-BODY-SEUL-OK
 = 2 : TRT-BODY-M/S-OK
 = 3 : TRT-GEN-BODY-OK
 = 4 : TRT-GERME/BOD-OK
 = 5 : TRT-PART-BODY-OK
 = 6 : TRT-GERME-CIBLE-OK
 = 7 : TRT-GERME-PATHO-OK

Sorties : Soit message erreur (MESSAGE EXTERNE)
 Soit

TRT-BODY-SEUL-OK/TRT-BODY-SEUL-LIST-OK (MESSAGE INTERNE)
 TRT-BODY-M/S-OK/TRT-BODY-M/S-LIST-OK (MESSAGE INTERNE)
 TRT-GEN-BODY-OK/TRT-GEN-BODY-LIST-OK (MESSAGE INTERNE)
 TRT-GERME/BODY-OK/TRT-GERME/BODY-LIST-OK (MES. INTERNE)
 TRT-PART-BODY-OK/TRT-PART-BODY-LIST-OK (MES. INTERNE)
 TRT-GERME-CIBLE-OK/TRT-GERME-CIBLE-LIST-OK (MES. INTERNE)
 TRT-GERME-PATHO-OK/TRT-GERME-PATHO-LIST-OK (MES. INTERNE)

PHASE 3 : CREATION/MAJ GERME

FONCTION MAJ-GERME

Entrée : TRT-GERME-OK (MESSAGE INTERNE)
 CODE-GERME (MESSAGE EXTERNE)
 GENRE-GERME (MESSAGE EXTERNE)
 ESPECE-GERME (MESSAGE EXTERNE)
 SEROTYPE-GERME (MESSAGE EXTERNE)
 FORME-GERME (MESSAGE EXTERNE)
 COLORATION-GRAM (MESSAGE EXTERNE)
 LIBELLE-FR-GERME (MESSAGE EXTERNE)
 LIBELLE-NL-GERME (MESSAGE EXTERNE)
 ...

Traitement : Saisie CODE-GERME
 Si CODE-GERME = ' '
 alors SUPP-GERME-OK
 sinon lire T.E GERME (CODE-GERME)
 si germe OK
 alors saisie info.germes
 validation info.germes
 si validation OK
 alors MAJ-GERME-FIN
 sinon message erreur
 sinon CRE-GERME-OK

Sorties : Soit message erreur (MESSAGE EXTERNE)
 Soit MAJ-GERME-FIN (MESSAGE EXTERNE)
 Soit CRE-GERME-OK (MESSAGE INTERNE)
 Soit SUPP-GERME-OK (MESSAGE INTERNE)

Ecran : Voir annexe

FONCTION CREATION-GERME

Entrée : CRE-GERME-OK (MESSAGE INTERNE)
CODE-GERME (MESSAGE EXTERNE)
GENRE-GERME (MESSAGE EXTERNE)
...
LIBELLE-NL-GERME (MESSAGE EXTERNE)

Traitement : Saisie informations (CODE-GERME, LIBELLE-FR-GERME,...)
Validation informations
si validation OK
alors CRE-GERME-FIN
sinon édition message d'erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
Soit CRE-GERME-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

FONCTION SUPPRESSION-GERME

Entrée : TRT-SUPP-GERME-OK (MESSAGE INTERNE)

Traitement : Suppression du germe
Si suppression correcte
alors SUPP-GERME-FIN
sinon message d'erreur

Sorties : Soit message d'erreur (MESSAGE EXTERNE)
Soit SUPP-GERME-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

PHASE 4 : EDITION GERME

FONCTION EDIT-GERME

Entrée : TERM-OK (MESSAGE INTERNE)
TRT-GERME-LIST-OK (MESSAGE INTERNE)
CODE-GERME (MESSAGE EXTERNE)

Traitement : Saisie CODE-GERME
 si CODE-GERME = '*'
 alors édition des informations concer-
 nant tous les germes
 EDIT-GERME-FIN
 sinon lecture T.E GERME (CODE GERME)
 si lecture OK
 alors édition informations de
 ce germe et
 EDIT-GERME-FIN
 sinon message erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
 Soit EDIT-GERME-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

PHASE 5 : CREATION/MAJ BODY-SITE

FONCTION MAJ-BODY-SITE

Entrée :

TRT-BODY-SEUL-OK	(MESSAGE INTERNE)
CODE-BODY-SITE	(MESSAGE EXTERNE)
LIBELLE-FR-BODY-SITE	(MESSAGE EXTERNE)
LIBELLE-NL-BODY-SITE	(MESSAGE EXTERNE)
LIBELLE-SSF-FR-BODY-SITE	(MESSAGE EXTERNE)
LIBELLE-SSF-NL-BODY-SITE	(MESSAGE EXTERNE)
LIBELLE-FAM-BODY-SITE	(MESSAGE EXTERNE)
DEF-NUM-BODY-SITE	(MESSAGE EXTERNE)
TYPE-NUM-BODY-SITE	(MESSAGE EXTERNE)
NUM-MIN-BODY-SITE	(MESSAGE EXTERNE)
NUM-MAX-BODY-SITE	(MESSAGE EXTERNE)

Traitement :

```
Saisie CODE-BODY-SITE
Si CODE-BODY-SITE = ' '
    alors SUPP-BODY-SITE-OK
    sinon lire body-site (CODE-BODY-SITE)
        si body-site OK
            alors saisie informations body-site
                validation info. body-site
                si validation OK
                    alors MAJ-BODY-SITE-FIN
                    sinon message erreur
        sinon CRE-BODY-SITE-OK
```

Sorties : Soit message erreur (MESSAGE EXTERNE)
Soit MAJ-BODY-SITE-FIN (MESSAGE EXTERNE)
Soit CRE-BODY-SITE-OK (MESSAGE INTERNE)
Soit SUPP-BODY-SITE-OK (MESSAGE INTERNE)

Ecran : Voir annexe

FONCTION CREATION-BODY-SITE

Entrée :

```
CRE-BODY-SITE-OK (MESSAGE INTERNE)
CODE-BODY-SITE (MESSAGE EXTERNE)
LIBELLE-FR-BODY-SITE (MESSAGE EXTERNE)
LIBELLE-NL-BODY-SITE (MESSAGE EXTERNE)
LIBELLE-SSF-FR-BODY-SITE (MESSAGE EXTERNE)
LIBELLE-SSF-NL-BODY-SITE (MESSAGE EXTERNE)
DEF-NUM-BODY-SITE (MESSAGE EXTERNE)
...
```

Traitement : Saisie informations (CODE, LIBELLE-FR-BODY-SITE,...)
Validation informations
si validation OK
alors CRE-BODY-SITE-FIN
sinon édition message d'erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
Soit CRE-BODY-SITE-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

FONCTION SUPPRESSION-BODY-SITE

Entrée : SUPP-BODY-SITE-OK (MESSAGE INTERNE)

Traitement : Suppression du germe
Si suppression correcte
alors SUPP-BODY-SITE-FIN
sinon message d'erreur

Sorties : Soit message d'erreur (MESSAGE EXTERNE)
Soit SUPP-BODY-SITE-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

PHASE 6 : EDITION BODY-SITE

FONCTION EDIT-BODY-SITE

Entrée : TERM-OK (MESSAGE INTERNE)
TRT-BODY-SITE-SEUL-LIST-OK (MESSAGE INTERNE)
CODE-BODY-SITE (MESSAGE EXTERNE)

Traitement :

Saisie CODE-BODY-SITE
Si CODE-BODY-SITE = '*'
alors édition des informations concernant tous les
body-sites
EDIT-BODY-SITE-SEUL-FIN
sinon lire body-site (CODE-BODY-SITE)
si body-site existe
alors édition info. de ce body-site et
EDIT-BODY-SITE-SEUL-FIN
sinon édition message erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
Soit EDIT-BODY-SITE-SEUL-FIN (MESS. INTERNE)

Ecran : Voir annexe

PHASE 7 : CREATION/MAJ-MASTER-SLAVE

FONCTION MAJ-MASTER-SLAVE

Entrée : TRT-BODY-M/S-OK (MESSAGE INTERNE)
BODY-SITE-MASTER (MESSAGE EXTERNE)
BODY-SITE-SLAVE (MESSAGE EXTERNE)

Traitement :

Saisie BODY-SITE-MASTER
lecture T.E BODY-SITE
si body-site = "master"
alors saisie BODY-SITE-SLAVE
lecture T.E BODY-SITE
si body-site = "slave"
alors MAJ-MASTER-SLAVE-FIN
sinon message d'erreur
sinon message d'erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
Soit MAJ-MASTER-SLAVE-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

PHASE 8 : CREATION/MAJ GENERALITES BODY-SITE GERMES

CONFONDUS

FONCTION MAJ-GENERALITES-BODY-SITE

Entrée : TRT-GEN-BODY-OK (MESSAGE INTERNE)
BODY-SITE-GENERALITES (MESSAGE EXTERNE)
INFORMATION-NUMERATION (MESSAGE EXTERNE)
NUMERATION-GENERALITE (MESSAGE EXTERNE)
DUREE-NON-REDONDANCE (MESSAGE EXTERNE)
ANTIBIOGRAMMES (MESSAGE EXTERNE)

Traitement :

Saisie BODY-SITE-GENERALITE

Lecture T.E BODY-SITE

Si body-site correct

alors lecture T.A GENERALITE-BODY-SITE

écho des informations connues

lecture informations introduites

validation des informations

si validation OK alors

si NUMERATION-GENERALITE = ' '

alors SUPP-GEN-BODY-OK

sinon mise à jour des infos

MAJ-GEN-BODY-FIN

sinon message erreur

sinon CRE-GEN-BODY-OK

sinon imprimer message d'erreur

sinon imprimer message d'erreur

Sorties	:	Soit message erreur	(MESSAGE EXTERNE)
		Soit MAJ-GEN-BODY-FIN	(MESSAGE EXTERNE)
		Soit CRE-GEN-BODY-OK	(MESSAGE INTERNE)
		Soit SUPP-GEN-BODY-OK	(MESSAGE INTERNE)

Ecran : Voir annexe

FONCTION CREATION-GENERALITES-BODY-SITE

Entrée :

CRE-GEN-BODY-OK	(MESSAGE INTERNE)
BODY-SITE-GENERALITE	(MESSAGE EXTERNE)
INFORMATION-GENERALITE	(MESSAGE EXTERNE)
DUREE-NON-REDONDANCE	(MESSAGE EXTERNE)
NUMERATION-GENERALITE	(MESSAGE EXTERNE)
ANTIBIOGRAMMES	(MESSAGE EXTERNE)

Traitement :

Saisie informations (BODY-SITE-GENERALITES,...)

Validation informations

si validation OK alors CRE-GEN-BODY-SITE-FIN

sinon édition message erreur

Sorties	:	Soit message erreur	(MESSAGE EXTERNE)
		Soit CRE-GEN-BODY-SITE-FIN	(MESSAGE EXTERNE)

Ecran : Voir annexe

Traitement :

```
Sorties      : Soit message erreur          (MESSAGE EXTERNE)
              Soit SUPP-GEN-BODY-SITE-FIN   (MESS. EXTERNE)
```

Ecran : Voir annexe

Traitement :

sinon édition message erreur

Ecran : Voir annexe

PHASE 10 : CREATION/MAJ PARTICULARITES-GERME-BODY-SITE

FONCTION MAJ-GERME-PARTICULIER-DANS-BODY-SITE

Entrée : TRT-GERME-BODY-OK (MESSAGE INTERNE)
 BODY-SITE-PARTICULIER (MESSAGE EXTERNE)
 GERME-PARTICULIER (MESSAGE EXTERNE)
 INFORMATION-NUMERATION (MESSAGE EXTERNE)
 NUMERATION-PARTICULIER (MESSAGE EXTERNE)
 DUREE-NON-REDONDANCE (MESSAGE EXTERNE)
 NUMERATION-PARTICULIER (MESSAGE EXTERNE)

Traitement :

Saisie BODY-SITE-PARTICULIER
 Lecture T.E BODY-SITE
 Si body-site existe
 alors lecture T.E GERMES
 Si germe existe ou GERME-PARTICULIER = '*'
 alors lecture T.A PARTICULARITE-GERME-BODY-SITE
 Si existe
 alors saisie informations
 validation info.
 si NUMERATION-PARTICULIER = ' '
 alors SUPP-PART-BODY-OK
 sinon MAJ-PART-BODY-FIN
 sinon message erreur
 sinon CRE-PART-BODY-OK
 sinon imprimer message d'erreur
 sinon imprimer message d'erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
 Soit MAJ-PART-BODY-FIN (MESSAGE EXTERNE)
 Soit CRE-PART-BODY-OK (MESSAGE INTERNE)
 Soit SUPP-PART-BODY-OK (MESSAGE INTERNE)

Ecran : Voir annexe

FONCTION CREATION-GERME-PART-BODY-SITE

Entrée : CRE-PART-BODY-OK (MESSAGE INTERNE)
 BODY-SITE-PARTICULIER (MESSAGE EXTERNE)
 GERME-PARTICULIER (MESSAGE EXTERNE)
 INFORMATION-NUMERATION (MESSAGE EXTERNE)
 NUMERATION-PARTICULIER (MESSAGE EXTERNE)
 DUREE-NON-REDONDANCE-PARTICULIER (MESSAGE EXTERNE)
 NUMERATION-PARTICULIER (MESSAGE EXTERNE)

Traitement :

Saisie informations (BODY-SITE-PARTICULIER,...)

Validation informations

 si validation OK

 alors CRE-PART-BODY-FIN

 sinon édition message d'erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
 : Soit CRE-PART-BODY-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

FONCTION SUPPRESSION-GERME-PARTICULIER-BODY

Entrée : SUPP-PART-BODY-OK (MESSAGE INTERNE)

Traitement :

Suppression d'une occurrence du T.A PARTICULARITE-GERME-BODY-SITE

Si suppression correcte alors SUPP-PART-BODY-FIN
 sinon message d'erreur

Sorties : Soit message d'erreur
 : Soit SUPP-PART-BODY-FIN (MESSAGE INTERNE)

Ecran : Voir annexe

PHASE 11 : EDITION PARTICULARITE-GERME-BODY-SITE

FONCTION EDIT-GERME-PARTICULIER-BODY-SITE

Entrée : TERM-OK (MESSAGE INTERNE)
 : TRT-PART-BODY-LIST-OK (MESSAGE INTERNE)
 : BODY-SITE-PARTICULIER (MESSAGE EXTERNE)
 : GERME-PARTICULIER (MESSAGE EXTERNE)

Traitement :

Lecture T.E BODY-SITE

si body-site existe

alors lecture T.A PARTICULARITE-GERME-BODY-SITE

si BODY-SITE-PARTICULIER = '*'

alors édition des informations concernant tous
les germes particuliers dans ces body-sites
EDIT-PART-BODY-FIN

sinon lire germe (GERME-PARTICULIER)

Si germe existe

alors édition informations sur ce germe
dans le(s) body-site(s) précisés.
EDIT-PART-BODY-FIN

sinon édition message erreur

sinon édition message erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
Soit EDIT-PART-BODY-FIN (MESSAGE EXTERNE)

PHASE 12 : CREATION/MAJ GERME-CIBLE

FONCTION MAJ-GERME-CIBLE

Entrée : TRT-GERME-CIBLE-OK (MESSAGE INTERNE)
BODY-SITE-CIBLE (MESSAGE EXTERNE)
GERME-CIBLE (MESSAGE EXTERNE)
NUMERATION-CIBLE (MESSAGE EXTERNE)
DUREE-RECH.-INFECTION-CROISSEE (M. EXTERNE)
ANTIBIOTIQUE-CIBLE (MESSAGE EXTERNE)
RESULTAT-ANTIBIOTIQUE-CIBLE (M. EXTERNE)

Traitement :

Saisie BODY-SITE-CIBLE

Lecture T.E BODY-SITE

Si body-site existe

alors lecture T.E GERMES

si germe existe alors lecture T.A GERMES-CIBLES

si existe alors saisie info's

si NUMERATION-CIBLE = ' '

alors SUPP-GERME-CIBLE-OK

sinon validation info's

si validation OK

alors mise-à-jour et

MAJ-G.C-FIN

sinon CRE-G.C-OK

sinon message d'erreur

sinon message d'erreur

sinon imprimer message d'erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
 Soit MAJ-G.C-FIN (MESSAGE EXTERNE)
 Soit SUPP-GERME-CIBLE-OK (MESSAGE INTERNE)
 Soit CRE-G.C-OK (MESSAGE INTERNE)

FONCTION CREATION-GERME-CIBLE

Entrée : CRE-G.C-OK (MESSAGE INTERNE)
 GERME-CIBLE (MESSAGE EXTERNE)
 NUMERATION-CIBLE (MESSAGE EXTERNE)
 DUREE-RECH-INFECTION-CROISEE (MESS. EXTERNE)
 ANTIBIOTIQUE-CIBLE (MESSAGE EXTERNE)
 RESULTAT-ANTIBIOTIQUE-CIBLE (MESS. EXTERNE)

Traitement :

Saisie informations (GERME-CIBLE, NUMERATION-CIBLE,...)

Validation informations

si validation OK

alors création d'un nouveau germe cible et

CRE-GERME-CIBLE-FIN

sinon édition message d'erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
 Soit CRE-GERME-CIBLE-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

FONCTION SUPPRESSION-GERME-CIBLE

Entrée : SUPP-GERME-CIBLE-OK (MESSAGE INTERNE)

Traitement : Suppression du germe cible et des informations qui y sont associées.

Si suppression correcte

alors SUPP-GERME-CIBLE-FIN

sinon message d'erreur

Sorties : Soit message d'erreur (MESSAGE EXTERNE)
 Soit SUPP-GERME-CIBLE-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

PHASE 13 : EDITION GERME-CIBLE

FONCTION EDIT-GERME-CIBLE

Entrée : TERM-OK (MESSAGE INTERNE)
TRT-GERME-CIBLE-LIST-OK (MESSAGE INTERNE)
GERME-CIBLE (MESSAGE EXTERNE)

Traitement :

Lecture T.A GERME-CIBLE
Si GERME-CIBLE = '*'
alors édition des informations concernant tous les
germes cibles connus
EDIT-GERME-CIBLE-FIN
sinon lire T.A GERME-CIBLE
si germe cible existe
alors édition informations sur ce germe cible
EDIT-GERME-CIBLE-FIN
sinon édition message erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
Soit EDIT-GERME-CIBLE-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

PHASE 14 : CREATION/MAJ GERMES PATHOGENES

FONCTION MAJ-GERME-PATHO

Entrée : TRT-GERME-PATHO-OK (MESSAGE INTERNE)
BODY-SITE-PATHOGENE (MESSAGE EXTERNE)
GERME-PATHOGENE (MESSAGE EXTERNE)
NUMERATION-PATHOGENE (MESSAGE EXTERNE)
DELAJ-EDITION-INFECTION-PATHO. (M. EXTERNE)
ANTIBIOTIQUE-PATHOGENE (MESSAGE EXTERNE)
RESULTAT-ANTIBIOTIQUE-PATHOGENE (M. EXTERNE)

Traitement :

```
Saisie BODY-SITE-PATHOGENE
Lecture T.E BODY-SITE
Si body-site existe
  alors lecture T.E GERMES
Si germe existe
  alors lecture T.A GERME-PATHOGENE
Si existe
  alors saisie info's
  Si NUMERATION-PATHOGENE = ' '
    alors SUPP-GERME-PATHO-OK
    sinon validation info's
      si validation ok
        alors mise-à-jour et
        MAJ-G.P-FIN
        sinon CRE-G.P-OK
      sinon message d'erreur
    sinon imprimer message d'erreur
```

Sorties	: Soit message erreur	(MESSAGE EXTERNE)
	Soit MAJ-GERME-PATHO-FIN	(MESSAGE EXTERNE)
	Soit SUPP-GERME-PATHO-OK	(MESSAGE INTERNE)
	Soit CRE-G.P-OK	(MESSAGE INTERNE)

Ecran : Voir annexe

FONCTION CREATION-GERME-PATHOGENE

Entrée	: CRE-GERME-PATHO-OK	(MESSAGE INTERNE)
	GERME-PATHOGENE	(MESSAGE EXTERNE)
	NUMERATION-PATHOGENE	(MESSAGE EXTERNE)
	DELAI-EDITION-INFECTION-PATHOGENE	(M. EX.)
	ANTIBIOTIQUE-PATHOGENE	(MESSAGE EXTERNE)
	RESULTAT-ANTIBIOTIQUE-PATHOGENE	(M. EXTERNE)

Traitement : Saisie informations (GERME-PATHOGENE,...)
Validation informations
si validation OK
alors CRE-GERME-PATHO-FIN
sinon édition message d'erreur

Sorties	: Soit message erreur	(MESSAGE EXTERNE)
	Soit CRE-GERME-PATHO-FIN	(MESSAGE EXTERNE)

Ecran : Voir annexe

FONCTION SUPPRESSION-GERME-PATHO

Entrée : SUPP-GERME-PATHO-OK (MESSAGE INTERNE)

Traitement : Suppression du germe dit "pathogène" dans ce
body-site, et des informations qui y sont
associées.
Si suppression correcte
alors SUPP-GERME-PATHO-FIN
sinon message d'erreur

Sorties : Soit message d'erreur (MESSAGE EXTERNE)
Soit SUPP-GERME-PATHO-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

PHASE 15 : EDITION GERME-PATHOGENES

FONCTION EDIT-GERME-PATHOGENES

Entrée : TERM-OK (MESSAGE INTERNE)
TRT-GERME-PATHO-LIST-OK (MESSAGE INTERNE)
GERME-PATHOGENE (MESSAGE EXTERNE)

Traitement :

Lecture T.A GERMES-PATHO.
si GERME-PATHO = '*'
alors édition des informations concernant tous les
germes pathogènes connus
EDIT-GERME-PATHO-FIN
sinon lire T.A GERME-PATHO
si germe pathogène existe
alors édition informations sur ce germe pathogène
et EDIT-GERME-PATHO-FIN
sinon édition message erreur

Sorties : Soit message erreur (MESSAGE EXTERNE)
Soit EDIT-GERME-PATHO-FIN (MESSAGE EXTERNE)

Ecran : Voir annexe

PHASE 16 : ACCEPTATION COMMANDES EPIDEMIOLOGIE

FONCTION SAISIE-CDE-EPI

Entrée : TYPE-CDE-EPI (MESSAGE INTERNE)

Traitement : Saisie TYPE-CDE-EPI
 Validation TYPE-CDE-EPI
 Si TYPE-CDE 'SOS', 'HELP', 'KB', 'TERM',
 'FRAN', 'NEDE', 'LIST', 'STOP',
 'BYE'
 alors message d'erreur
 sinon case 'SOS' : TRT-SOS-OK
 'HELP' : TRT-HELP-OK
 'KB' : TRT-KB-OK
 'TERM' : TRT-TERM-OK
 'FRAN' : TRT-FRAN-OK
 'NEDE' : TRT-NEDE-OK
 'LIST' : TRT-LIST-EPI-OK
 'STOP' : TRT-STOP-OK
 'BYE' : TRT-BYE-OK

Sorties : Soit message erreur (OUTPUT)
 TRT-LIST-EPI-OK (MESSAGE INTERNE)
 TRT-TERM-EPI-OK (MESSAGE INTERNE)
 ...

FONCTION SOS

Idem FONCTION SOS de la phase 1.

FONCTION HELP

Idem FONCTION HELP de la phase 2.

FONCTION KB

Idem FONCTION KB de la phase 2.

FONCTION TERM

Idem FONCTION TERM de la PHASE 2.

FONCTION FRAN

Idem FONCTION FRAN de la PHASE 2.

FONCTION NEDE

Idem FONCTION NEDE de la PHASE 2.

FONCTION STOP

Idem FONCTION STOP de la PHASE 2.

FONCTION BYE

Idem FONCTION BYE de la PHASE 2.

FONCTION SAISIE-RUBRIQUE-LIST-EPI

Entrée	:	NUMERO-RUBRIQUE	(MESSAGE EXTERNE)
		TRT-LIST-EPI-OK	(MESSAGE INTERNE)

Traitement :

Saisie NUMERO-RUBRIQUE
Validation NUMERO-RUBRIQUE
Si NUMERO-RUBRIQUE 1 ou 3
 alors imprimer message erreur
 sinon case NUMERO-RUBRIQUE = 1 : TRT-LIST-JRN-OK
 " = 2 : TRT-LIST-INF-OK
 " = 3 : TRT-STAT-EPI-OK

Sorties	:	Soit message d'erreur	(MESSAGE EXTERNE)
		Soit TRT-LIST-JRN-OK	(MESSAGE INTERNE)
		TRT-LIST-INF-OK	(MESSAGE INTERNE)
		TRT-STAT-EPI-OK	(MESSAGE INTERNE)

Ecran : Voir annexe

FONCTION SAISIE-RUBRIQUE-LIST-INF-EPI

Entrée : NUMERO-RUBRIQUE (MESSAGE EXTERNE)
 TRT-LIST-INF-OK (MESSAGE INTERNE)

Traitement :

Saisie NUMERO-RUBRIQUE
 Validation NUMERO-RUBRIQUE
 Si NUMERO-RUBRIQUE 1 ou 2
 alors imprimer message erreur
 sinon case NUMERO-RUBRIQUE = 1 : TRT-LIST-INF-1-OK
 " = 2 : TRT-LIST-INF-2-OK

Sorties : Soit message d'erreur (MESSAGE EXTERNE)
 Soit TRT-LIST-INF-1-OK (MESSAGE INTERNE)
 Soit TRT-LIST-INF-2-OK (MESSAGE INTERNE)

Ecran : Voir annexe

FONCTION SAISIE-RUBRIQUE-STAT-EPI

Entrée : NUMERO-RUBRIQUE (MESSAGE EXTERNE)
 TRT-STAT-EPI-OK

Traitement :

Saisie NUMERO-RUBRIQUE
 Validation NUMERO-RUBRIQUE
 Si NUMERO-RUBRIQUE 1 ou 5
 alors imprimer message erreur
 sinon case NUMERO-RUBRIQUE = 1 : TRT-REPART-GERME-SERV-OK
 " = 2 : TJRT-TAUX-INF-SERV-OK
 " = 3 : TRT-TAUX-INF-BODY-SERV-OK
 " = 4 : TRT-HISTOGRAMME-OK
 " = 5 : TRT-POLYRESISTANTS-OK

Sorties :

Soit message erreur (OUTPUT)
 TRT-REPART-GERME-SERV-OK (MESSAGE INTERNE)
 TRT-TAUX-INF-SERVICE-OK (MESSAGE INTERNE)
 TRT-TAUX-INF-BODY-SERV-OK (MESSAGE INTERNE)
 TRT-HISTOGRAMME-OK (MESSAGE INTERNE)
 TRT-POLYRESISTANT-OK (MESSAGE INTERNE)

Ecran : Voir annexe

PHASE 17 : EDITION LISTES TRIEES

FONCTION SAISIE-DONNEES

Entrée :

TRT-LIST-INF-1-OK/TRT-LIST-INF-2-OK	(MESSAGE INTERNE)
PERIODE	(MESSAGE EXTERNE)
CODE-GERME	(MESSAGE EXTERNE)
CODE-SERVICE	(MESSAGE EXTERNE)

Traitement : Validation des données introduites

Sortie	:	Soit message erreur	(MESSAGE EXTERNE)
		Soit PERIODE-OK	(MESSAGE INTERNE)
		CODE-GERME-OK	(MESSAGE INTERNE)
		CODE-SERVICE-OK	(MESSAGE INTERNE)

FONCTION SELECTION

Entrée :

TRT-LIST-INF-1-OK/TRT-LIST-INF-2-OK	(MESSAGE INTERNE)
PERIODE-OK	(MESSAGE INTERNE)
CODE-GERME-OK	(MESSAGE INTERNE)
CODE-SERVICE-OK	(MESSAGE INTERNE)

Traitement :

Sélection des résultats non redondants sur base de la
table de décision définie
dans la description de la solution projetée
(si CODE-GERME = '*' alors ne pas tenir compte de la
colonne correspondante; idem pour CODE-SERVICE)

Sorties	:	CANDIDATS-SELECTIONNES	(MESSAGE INTERNE)
---------	---	------------------------	-------------------

FONCTION TRI

Entrée :

TRT-LIST-INF-1-OK/TRT-LIST-INF-2-OK (MESSAGE INTERNE)
CANDIDATS-SELECTIONNES (MESSAGE INTERNE)

Traitement :

Tri des CANDIDATS-SELECTIONNES suivant la clef :
CODE-GERME+CODE-SERVICE+NOM-PATIENT+DATE-DEM. pour liste 1
CODE-SERVICE+CODE-GERME+NOM-PATIENT+DATE-DEM. pour liste 2

Sorties : CANDIDATS-SELECTIONNES-TRIES (MESS. INTERNE)

FONCTION EDITION

Entrée : CANDIDATS-SELECTIONNES-TRIES (MESS. INTERNE)

Traitement : Lecture CANDIDATS-SELECTIONNES-TRIES et
édition.

Sorties : Liste 1 / liste 2 (MESSAGE EXTERNE)

PHASE 18 : EDITION/AFFICHAGE STATISTIQUES

FONCTION SAISIE-DONNEES-STAT-1

Entrées : TRT-REPART-GERME-SERV-OK (MESSAGE INTERNE)
PERIODE (MESSAGE EXTERNE)
CODE-GERME (MESSAGE EXTERNE)

Traitement : Validation des données introduites

Sorties : Soit message erreur (MESSAGE EXTERNE)
PERIODE-OK (MESSAGE INTERNE)
CODE-GERME-OK (MESSAGE INTERNE)
CODE-SERVICE-OK (MESSAGE INTERNE)

FONCTION SELECTION-TRI-EDITION-STAT-1

Entrée : PERIODE-OK (MESSAGE INTERNE)
CODE-GERME-OK (MESSAGE INTERNE)
CODE-SERVICE-OK (MESSAGE INTERNE)

Traitement :

Obtention des CANDIDATS-STAT-1 par sélection des résultats non redondants.

Sur base de la table de décision définie dans la description de la solution projetée.

(Si CODE-GERME = '*'
alors tenir compte de tous les germes rencontrés;
si CODE-GERME = '*'
alors tenir compte de(s) germe(s) rencontré(s) dans
tous les services hospitaliers).

Tri des CANDIDATS-STAT-1 sur base de la clef :
CODE-SERVICE+CODE-GERME avec comptage des germes
identiques rencontrés dans le même service et calcul du
nombre de germes sélectionnés au total.

Edition des CANDIDATS-SELECTIONNES-TRIES-STAT-1 avec
calcul du rapport entre le nombre de germes identiques
par service et le nombre total de germes rencontrés.

Sorties : LISTER-STAT-1-FIN (MESSAGE EXTERNE)

Liste : Voir annexe

FONCTION SAISIE-DONNEES-STAT-2

Entrées : TRT-TAUX-INF-SERVICE-OK (MESSAGE INTERNE)
PERIODE (MESSAGE EXTERNE)
CODE-SERVICE (MESSAGE EXTERNE)

Traitement : Validations des données introduites.

Sorties : Soit message erreur (MESSAGE EXTERNE)
PERIODE-OK (MESSAGE INTERNE)
CODE-SERVICE-OK (MESSAGE INTERNE)

FONCTION SELECTION-TRI-EDITION-STAT-2

Entrée : PERIODE-OK (MESSAGE INTERNE)
CODE-SERVICE-OK (MESSAGE INTERNE)

Traitement :

Obtention des CANDIDATS-STAT-2 par sélection des résultats non redondants.

Sur base de la table de décision définie dans la description de la solution projetée.

(Si CODE-SERVICE = '*')

alors tenir compte de(s) germe(s) rencontré(s) dans tous les services hospitaliers).

Tri des CANDIDATS-STAT-2 sur base de la clef :

CODE-SERVICE+CODE-BODY-SITE+CODE-GERME avec comptage du nombre de germes rencontrés par body-site dans chaque service et le calcul du nombre de body-site sélectionnés quel que soit le résultat.

Edition des CANDIDATS-SELECTIONNES-TRIES-STAT-2 avec calcul du rapport entre le nombre de germes rencontrés par body-site dans chaque service et le nombre total de body-sites quel que soit le résultat.

Sorties : LISTE-STAT-2-FIN (MESSAGE EXTERNE)

Liste : Voir annexe

FONCTION SAISIE-DONNEES-STAT-3

Entrées : TRT-TAUX-INF-BODY-SRV-OK (MESSAGE INTERNE)
PERIODE (MESSAGE EXTERNE)
CODE-BODY-SITE (MESSAGE EXTERNE)
CODE-SERVICE (MESSAGE EXTERNE)

Traitement : Validations des données introduites.

Sorties : Soit message erreur (MESSAGE EXTERNE)
PERIODE-OK (MESSAGE INTERNE)
CODE-SERVICE-OK (MESSAGE INTERNE)
CODE-BODY-SITE-OK (MESSAGE INTERNE)

FONCTION SELECTION-TRI-EDITION-STAT-3

Entrée : PERIODE-OK (MESSAGE INTERNE)
CODE-SERVICE-OK (MESSAGE INTERNE)
CODE-BODY-SITE-OK (MESSAGE INTERNE)

Traitement :

Obtention des CANDIDATS-STAT-3 par sélection des résultats non redondants.

Sur base de la table de décision définie dans la description de la solution projetée.

(Si CODE-BODY-SITE = '*')

alors tenir compte de tous les body-sites rencontrés
si CODE-SERVICE = '*'

alors tenir compte des body-sites rencontrés dans
tous les services hospitaliers).

Tri des CANDIDATS-STAT-3 sur base de la clef :

CODE-SERVICE+CODE-BODY-SITE+CODE-GERME avec comptage du
nombre de germes identiques rencontrés par body-site dans
chaque service et calcul du nombre de body-site
sélectionnés quel que soit le résultat.

Edition des CANDIDATS-SELECTIONNES-TRIES-STAT-3 avec
calcul du rapport entre le nombre de germes identiques
rencontrés par body-site et par service, et le nombre
total de body-sites quel que soit le résultat.

Sorties : LISTE-STAT-3-FIN (MESSAGE EXTERNE)

Liste : Voir annexe

FONCTION SAISIE-DONNEES-STAT-4

Entrées : TRT-HISTOGRAMME-OK (MESSAGE INTERNE)
PERIODE (MESSAGE EXTERNE)
CODE-GERME (MESSAGE EXTERNE)
N * CODE-ANTIBIOTIQUE (MESSAGE EXTERNE)

Traitement : Validations des données introduites.

Sorties : Soit message erreur (MESSAGE EXTERNE)
PERIODE-OK (MESSAGE INTERNE)
CODE-GERME-OK (MESSAGE INTERNE)
N * CODE-ANTIBIOTIQUE-ok (MESSAGE INTERNE)

FONCTION SELECTION-TRI-EDITION-STAT-4

Entrée : PERIODE-OK (MESSAGE INTERNE)
CODE-GERME-OK (MESSAGE INTERNE)
N * CODE-ANTIBIOTIQUE-OK (MESSAGE INTERNE)

Traitement :

Obtention des CANDIDATS-STAT-4 par sélection des résultats positifs.

Sur base de la table de décision définie dans la description de la solution projetée.

(Si CODE-GERME = '*')

alors tenir compte de tous les germes rencontrés)

Tri des CANDIDATS-STAT-4 sur base de la clef :

CODE-GERME+CODE-ANTIBIOTIQUE+STATUT-DEM avec comptage du nombre de germes identiques sensibles au même antibiotique et calcul du nombre de germes identiques rencontrés, en distinguant les résultats par le statut de la demande.

Editon des CANDIDATS-SELECTIONNES-TRIES-STAT-4 avec calcul des pourcentages de sensibilité.

Sorties : LISTE-STAT-4-AN (MESSAGE EXTERNE)

Liste : Voir annexe

FONCTION SAISIE-DONNEES-STAT-5

Entrées : TRT-POLYRESISTANT-OK (MESSAGE INTERNE)

PERIODE (MESSAGE EXTERNE)

N * CODE-ANTIBIOTIQUE (MESSAGE EXTERNE)

Traitement : Validations des données introduites

Sorties : Soit message erreur (MESSAGE EXTERNE)

PERIODE-OK (MESSAGE INTERNE)

N * CODE-ANTIBIOTIQUE-OK (MESSAGE INTERNE)

FONCTION SELECTION-TRI-EDITION-STAT-5

Entrée : PERIODE-OK (MESSAGE INTERNE)

N * CODE-ANTIBIOTIQUE (MESSAGE INTERNE)

Traitement :

Obtention des CANDIDATS-STAT-5 par sélection des résultats non redondants positifs.

Sur base de la table de décision définie dans la description de la solution projetée.

Tri des CANDIDATS-STAT-5 sur base de la clef :

CODE-GERME+CODE-ANTIBIOTIQUE avec comptage du nombre de germes identiques résistants à tous les antibiotiques et

calcul du nombre de germes identiques rencontrés, pour
lesquels tous les antibiotiques ont été testés.

Edition des CANDIDATS-SELECTIONNES-TRIES-STAT-5 avec
calcul des pourcentages de résistance.

Sorties : LISTE-STAT-5-FIN (MESSAGE EXTERNE)

Liste : Voir annexe

PHASE 19 : TRAITEMENT JOURNALIER

FONCTION-SELECTION-DONNEES-JRN

Entrées : TRT-LIST-JRN-OK (MESSAGE INTERNE)

Traitement :

Tri des résultats bactériologiques du jour sur base de la
clef de tri :

NUM-DEM-JRN+BODY-SITE-DEM-JNR+GERME-DEM-JRN+DATE-WRITE-
DEM-JRN+HEURE-WRITE-DEM-JRN

Elimination des résultats pour lesquels le CODE-GERME-
DEM-JRN est non définitif.

Sélection de la dernière version des résultats du jour
qui correspondent au même germe, même body-site et même
demande.

Sorties : RESULTATS-SELECTIONNES-JRN (MESSAGE INTERNE)

FONCTION TEST-REDONDANCE

Entrée : RESULTATS-SELECTIONNES-JRN (MESSAGE INTERNE)

Traitement :

Pour chaque RESULTATS-SELECTIONNES-JRN

si GERME-DEM-JRN 0

alors lecture délai de redondance

lecture T.E EPIDEMIOLOGIE pour le même patient
ayant même germe, même body-site

si (DATE-DEM-JRN - DATE-DERN-MERE-EPIDEMIO)

délai redondance

alors REDOND = FALSE

sinon lecture T.E EPIDEMIOLOGIE pour trouver
un résultat négatif pour ce patient et
ce body-site, situe entre DATE-DERN-
MERE-EPIDEMIO et DATE-DEM-JRN

si trouve

alors REDOND = FALSE

sinon REDOND = TRUE

sinon REDOND = FALSE

Ecriture résultats journaliers dans T.E EPIDEMIOLOGIE

Sorties : WRITE-EPIDEMIO-OK (MESSAGE INTERNE)

FONCTION TEST-PATHO

Entrée : WRITE-EPIDEMIO-OK (MESSAGE INTERNE)

Traitement :

Lire RESULTATS-SELECTIONNES-JRN

Si RESULTATS-SELECTIONNES-JRN hospitalisés

alors si CODE-GERME-DEM-JRN 0

alors lecture T.A GERME-PATHOGENE

test numération

si numération OK

alors test ABG

si test ABG OK

alors PATHO = TRUE

sinon PATHO = FALSE

sinon PATHO = FALSE

sinon PATHO = FALSE

sinon PATHO = FALSE

Mise-à-jour T.E EPIDEMIOLOGIE

Edition des résultats pathogènes dont la mère n'a pas
encore été éditée.

Sorties : LIST-PATHO-FIN (MESSAGE EXTERNE)

Listes : Voir annexe

FONCTION TEST-CIBLE

Entrée : LIST-PATHO-OK (MESSAGE INTERNE)
RESULTATS-SELECTIONNES-JRN (MESSAGE INTERNE)

Traitement :

Lire RESULTATS-SELECTIONNES-JRN
si RESULTATS-SELECTIONNES-JRN hospitalisés depuis au moins
48 h
alors si GERME-DEM-JRN 0
alors lecture T.A GERME-CIBLE
test numération
si numération OK
alors test ABG
si test ABG OK
alors CIBLE = TRUE
sinon CIBLE = FALSE
sinon CIBLE = FALSE
sinon CIBLE = FALSE
sinon CIBLE = FALSE
Mise-à-jour T.E EPIDEMIOLOGIE
Si CIBLE = TRUE alors recherche des personnes encore
hospitalisées actuellement depuis au moins 48 h qui ont
contracté le même germe.

Edition des résultats cibles avec leurs candidats
contaminants si la "mère" correspondante n'est pas encore
éditée.

Sorties : LIST-CIBLE-OK (MESSAGE INTERNE)

Listes : Voir annexe

VI. ANALYSE ORGANIQUE

1. PRESENTATION DES OUTILS UTILISES

1.1. Hardware

1.1.1. Equipement

Processeurs :

L'environnement dont nous disposons se compose de quatre miniordinateurs de type DIGITAL : - 2 VAX 785
- 2 PDP 11/73

Contrôleurs :

Les processeurs VAX sont équipés de deux contrôleurs disques. Chacun de ces contrôleurs gère quatre unités disques magnétiques.

Unités disques magnétiques :

Chaque unité disque peut être connectée à deux processeurs. Les unités disques sont identifiées par les numéros de contrôleur et leur numéro d'unité physique.

Unités bandes magnétiques :

Deux unités par processeur VAX
Deux unités pour les deux PDP.

1.1.2. Fiches techniques

VAX-785

- capacité mémoire centrale : 8 MB
- capacité mémoires auxiliaires : 1500 MB

PDP 11/73

- capacité mémoire centrale : 2 MB
- capacité mémoires auxiliaires : 350 MB

1.1.3. Configuration

Les terminaux sont reliés aux processeurs via un réseau de lignes téléphoniques. On compte environ 300 terminaux reliés à ce système.

1.2. Software de base

1.2.1. Système d'exploitation

La clinique Saint-Jean a choisi BOS (BASIC OPERATING SYSTEM) comme système d'exploitation pour leurs développements informatiques.

Cet operating system a été créé par le département data processing de l'université de Leiden, afin de satisfaire leurs besoins dans le cadre du développement d'un système intégré de gestion hospitalière, en collaboration avec différents hôpitaux universitaires hollandais.

Les caractéristiques principales de ce système sont :

- accès rapide aux données
- économie d'utilisation de l'espace mémoire
- sécurité des données
- facilité de maintenance, de réorganisation, d'extensibilité de la data base
- facilité d'utilisation
- utilisation du PDP 11/45, 11/70, 11/73, VAX et microVAX reliés sur un réseau ETHERNET
- possibilité de programmer en FORTRAN et en PASCAL (en cours d'adaptation)

La conception du système repose sur deux principes :

- la totale intégration des données
- la gestion de l'autorisation d'accès

1.2.2. Organisation des données

L'organisation des données se base sur quatre types de fichiers :

- les thesauri
- les TVR
- les random files
- les autres types de fichiers

1.2.2.1. Thesauri

Un thesaurus est un fichier qui contient un ensemble de record de longueur fixe, séquentiellement numéroté. Le type d'information est le même dans chaque thesaurus. Chaque record peut être accédé par son numéro de record, ou à partir de un ou plusieurs index, dérivés du contenu du record.

Chaque thesaurus est protégé contre les mises-à-jour

simultanées d'un même record et chaque index consiste en un ensemble de blocs logiques sur disque (7 index maximum).

Les blocs de chaque index sont triés par rapport à la clef et chaque bloc comprend un nombre variable de "bloc record index" non triés, de longueur fixe. Le contenu de chaque "bloc record index" est le numéro de record séquentiel thesaurus.

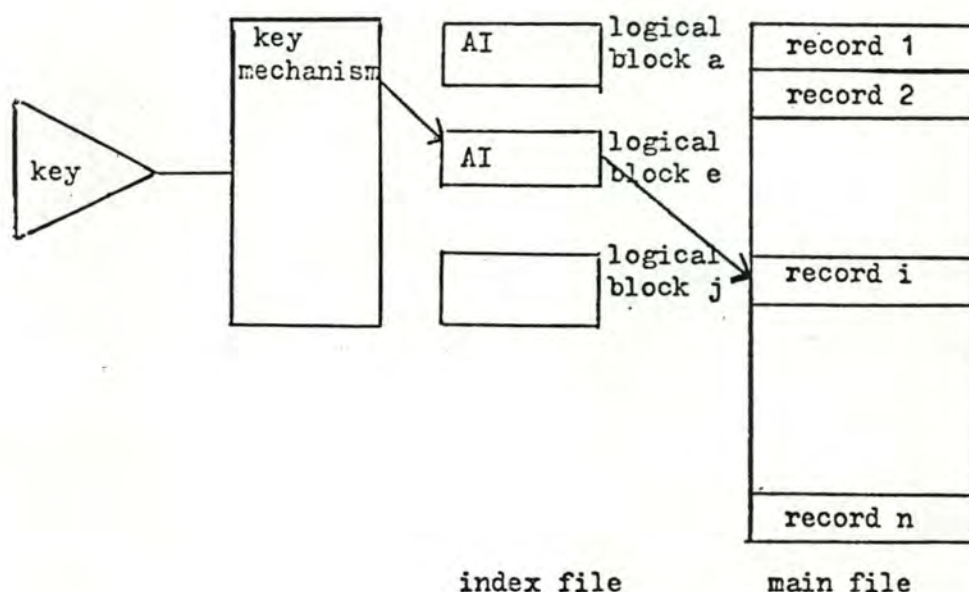
A la clef d'index est associée un ADI (Additional Discrimination Information) qui facilite et rend la recherche plus souple et plus efficace.

Le mécanisme de la clef utilisé pour les index est :

- soit du type hashing
- soit basé sur des tables d'index.

Exemple :

Une table est utilisée par les index du thesaurus des communes, et un mécanisme de hashing est utilisé pour l'index "date de naissance" du thesaurus des patients.



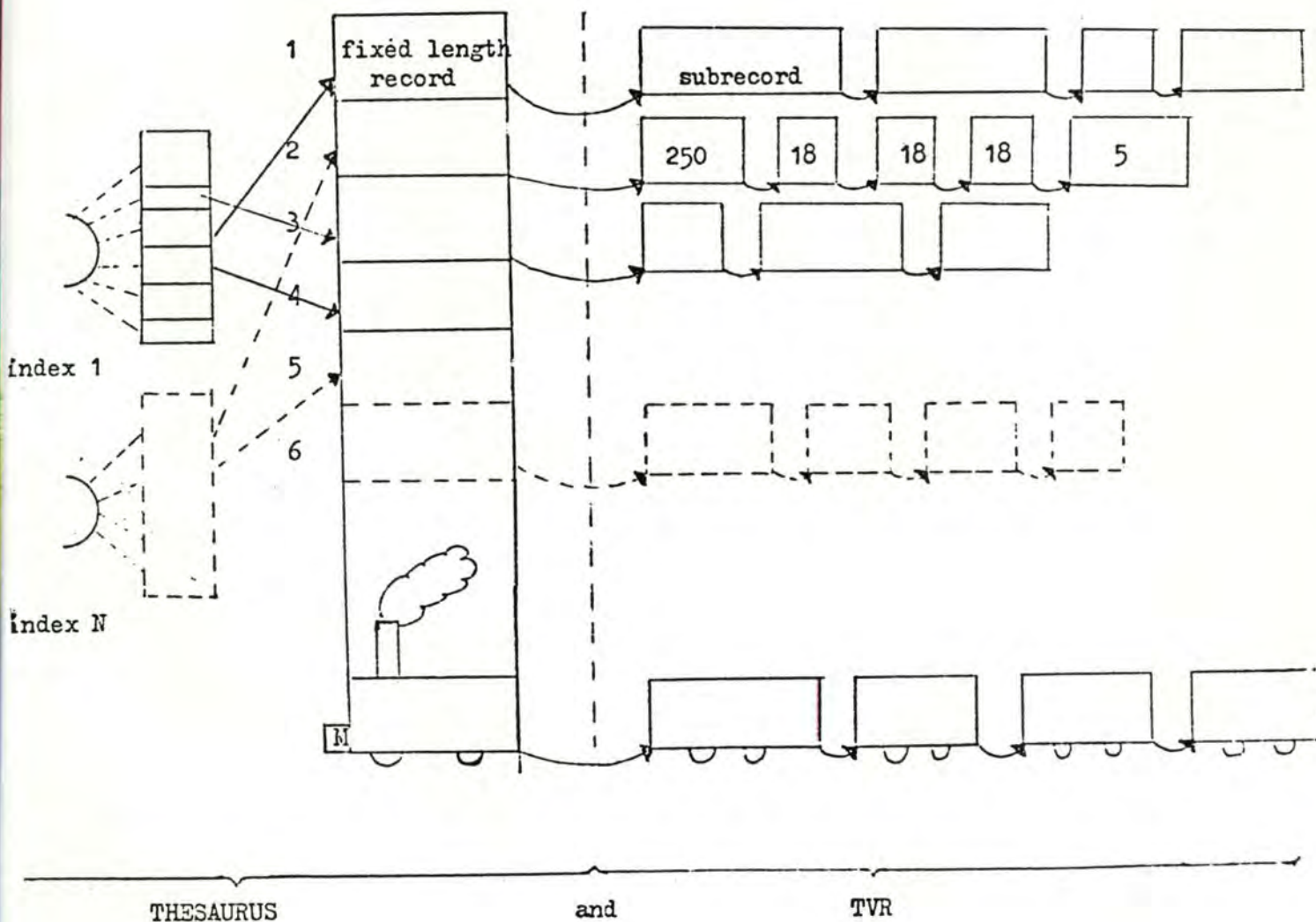
1.2.2.2. TVR

Un record de longueur variable peut être annexé à un record de thesaurus. Ces TVR consistent en une collection de type d'information distinct.

Un exemple est les diagnostics médicaux, les résultats d'analyses labo, qui peuvent être mémorisés dans des subrecords séparés.

Chaque TVR est identifié par un numéro et peut contenir un ou plusieurs sub-records de longueur variable.

Cette organisation de thesauri et de record en longueur variable (TVR) peut facilement être illustrée par le dessin ci-dessous :



1.2.2.3. Random files

Les fichiers random sont similaires aux thesauri puisqu'ils consistent en un ensemble de records de longueur fixe, séquentiellement numérotés.

Les différences essentielles sont :

- la non protection du fichier random vis-à-vis des updates multiples
- l'impossibilité d'implémenter des TVR sur les fichiers random
- l'impossibilité d'implémenter des index

Ces random-files sont appelés fichiers .R

1.2.2.4. Autres fichiers

On distingue cinq autres types de fichiers :

- Fichiers .D ou fichiers librairies créés à partir d'un workfile
- Fichiers .L qui sont les programmes utilisables
- Fichiers .S idem .R avec format et longueur variable
- Fichiers .T ou fichiers temporaires qui sont des workfiles existant uniquement pendant l'exécution du programme qui les a défini
- Fichiers .F qui contiennent la description des écrans

1.2.2.5. Taille des programmes

Au départ, le système était conçu pour être un ensemble de commandes et non de programmes. Chacune de ces commandes ne devait pas dépasser 16 K de mémoire. Cela facilitait le partage multi-users. Peu à peu, des programmes ont été écrits et nécessitaient plus de 16 K. Aussi, est-il nécessaire de décomposer les programmes écrits sous BOS en modules de 16 K.

2. SCHEMA DES ACCES POSSIBLES

2.1. Méthodologie

Le schéma des accès possibles exprime une structure de données qui est dotée d'une seule propriété : elle est la représentation aussi directe que possible du schéma conceptuel.

Son objectif est d'offrir une expression des données dans un formalisme adapté aux processus de conception des traitements.

Le schéma des accès possibles est très facilement représenté. A chaque type d'entité est associé un type d'article et à chaque propriété d'un type d'entité est associé un item.

A chaque type d'association binaire sans attribut est associé un type de chemin. A chaque type d'association binaire avec attribut correspond un type d'article, des items et deux types de chemins identifiant chaque type d'association.

2.2. Schéma des accès possibles

Lors de l'analyse de l'application épidémiologie, a été réalisé le schéma des accès possibles à partir du schéma conceptuel. Toutefois, afin de ne pas alourdir la présentation de ce travail, nous avons préféré détailler le schéma des accès nécessaires que nous avons déduit du schéma des accès possibles en adaptant celui-ci à l'organisation des données sous BOS.

3. ARCHITECTURE DES TRAITEMENTS

3.1. Méthodologie

Les spécifications fonctionnelles des phases et fonctions sont la base d'une architecture logicielle.

Il s'agit d'établir des relations entre les modules afin de définir une structure.

Pour restreindre les relations possibles, on organise le système en un ensemble de niveaux différents et ordonnés. On aboutit à une hiérarchie.

Il existe autant de hiérarchies que de sémantiques de la

relation établie entre les niveaux.

Pour cette application, nous avons adoptés le type de relation "UTILISE".

Définition : Si $R(A,B) : A \text{ "UTILISE" } B$
alors le fonctionnement correcte de A dépend
de la disponibilité d'une implémentation
correcte de B.

3.2. Architecture de traitements

On distingue dans la hiérarchie "utilise" sept niveaux :

NIVEAU 7 : Coordinateur des phases par type d'utilisateurs

NIVEAU 6 : Coordinateur des fonctions pour chaque phase

NIVEAU 5 : Noyau fonctionnel
Il s'agit de l'ensemble des modules fonctionnels
de base utilisés par les modules de NIVEAU 6

NIVEAU 4 : Modules utilitaires

NIVEAU 3 : Module de gestion des écrans et de gestion des
données

NIVEAU 2 : BOSFORM (gestionnaire d'écran sous BOS)
Gestionnaire thesaurus/tvr

NIVEAU 1 : BOS

3.3. Description des différents niveaux

Le but principal lors de la rédaction de ce mémoire a été d'une part de présenter une analyse conceptuelle et fonctionnelle complète, et d'autre part de réaliser la programmation et l'implémentation de ce projet épidémiologie. Dès lors, dans le cadre de cette analyse organique, nous préférons vous présenter la modularisation des traitements telle qu'elle a été réalisée en tenant compte des contraintes du système qui sont notamment la programmation en 16 K module et la difficulté de gérer l'accès aux données dans un environnement thesaurus/tvr.

Cette modularisation sera représentée par un schéma en annexe, lequel fera référence à différents exemples de modules documentés le plus complètement possible.

4. SCHEMA DES ACCES NECESSAIRES

4.1. Méthodologie

Sur base du schéma des accès possibles que nous avons réalisé lors de l'analyse organique, le schéma des accès nécessaires a été obtenu sur base de la description en pseudo-langage des phases et fonctions.

La fréquence d'utilisation et le type d'accès à chaque type d'article détermine l'organisation des fichiers.

4.2. Organisation des fichiers implémentés

La structure des données telle qu'elle nous est proposée par le système BOS, permet des enregistrements de longueur fixe ou variable, des accès séquentiels ou par clefs.

Les clefs sont primaires : identifiante non modifiable
secondaires : identifiante ou non
modifiable ou non

On peut décrire les clefs d'accès secondaires de manière très souple en créant des types d'index différents sur le même fichier.

4.3. Schéma des accès nécessaires

Tout type d'association avec attribut dans le schéma conceptuel a été transformé lors de la réalisation des schémas possibles en type d'article dont la clef d'accès est constituée par les identifiants des entités qu'elle reliait.

Il s'agit en fait, dans les termes et les notions que nous utilisons, d'un exemple parfait de TVR. En effet, le thesaurus qui porte le TVR constitue la première partie de la clef et une partie du contenu du TVR constitue la deuxième partie de la clef. Dans notre organisation, chaque type d'article devient un thesaurus.

La description du schéma des accès nécessaires se trouve en annexe.

5. CHOIX DES LANGAGES

- * La déclaration des nouvelles données et des nouveaux fichiers dans l'application bactériologie ont été développés en langage PASCAL/BOS. Cette décision a été prise étant donné la tendance du service informatique de la clinique de développer de plus en plus en PASCAL et de petit à petit quitter le FORTRAN qui est utilisé depuis la création du service informatique.

Il s'agit donc ici, d'un tout premier essai de développement dans ce langage à partir d'un nouvel outil de développement : PMAINT.

A partir des descriptions des écrans et des records, cet utilitaire crée automatiquement le programme de gestion correspondant. Quelques modifications ont cependant dû être apportées dans ces programmes.

- * La déclaration des données et la description des fichiers de l'application CONTROLE DES INFECTIONS, ainsi que les programmes de gestion et d'exploitation de ces données, ont été écrites en FORTRAN/BOS.

La raison principale de ce choix se situe dans l'intégration de cette nouvelle application dans une infrastructure complexe développée en FORTRAN/BOS. Développer la nouvelle application CONTROLE DES INFECTIONS en PASCAL aurait été très difficile et peu fructueuse étant donné "la jeunesse" de ce langage et le peu d'expérience vécue dans cet environnement.

PARTIE 4 : CONCLUSIONS ET PERSPECTIVES

I. CONCLUSIONS

Dans la première partie de notre travail, nous avons tenté de cerner le plus précisément possible les besoins réels de la clinique au niveau du contrôle des infections.

Cette étude nous a amené à revoir complètement la bactériologie pour laquelle aucune structure n'était compatible avec une quelconque statistique.

Actuellement, les techniciens comme le bactériologiste sont satisfaits de la nouvelle application. D'ailleurs la phase progressive d'implémentation de celle-ci a été plus rapide que prévu.

L'édition des listes d'infections pathologiques et croisées sont actuellement soumises au bactériologiste et exigeront très certainement de celui-ci une adaptation des germes cibles et pathogènes vu l'aspect informel de certaines données telle que le délai de redondance, la numération, les antibiotiques à tester, etc.

La description de la solution, l'analyse fonctionnelle, l'organique et la phase importante de programmation de ce projet m'a permis de connaître toute l'évolution d'un projet depuis sa conception jusqu'à la réalisation et l'implémentation sur le terrain.

Le succès de l'ensemble du projet dépend de la bonne participation des secrétaires à l'encodage précis des demandes, des techniciens au moment de l'introduction des résultats, du médecin bactériologiste dans la définition de la pathogénicité et du caractère cible des germes, et de la prise en charge par l'infirmier épidémiologiste des cas présentés dans les listes journalières.

II. PERSPECTIVES

- * Vu les structures simples prises dans la nouvelle solution, il est envisageable de pouvoir créer de nouvelles listes très facilement sans devoir modifier les programmes. Une liste de critère de sélection serait proposée, ainsi que différentes clefs de tri possibles.
- * Etant donné l'intégration totale des différents projets de la clinique, il semblerait très intéressant d'étudier l'évolution du taux de résistance de certains germes en fonction de la consommation de certains antibiotiques diffusés par la pharmacie.
- * A l'heure des microordinateurs, il serait agréable de pouvoir éditer graphiquement les résultats des statistiques épidémiologiques sur terminal de couleur à haute résolution graphique.
- * Parallèlement aux fichiers des germes, on peut imaginer l'implémentation de fichiers de virus et ainsi permettre une étude plus poussée en virologie.
- * L'implication des infections nosocomiales se manifeste au niveau de la durée du séjour du patient dans la clinique et de ce fait, augmente le coût du séjour.

Dès lors, on peut imaginer dans un avenir proche la demande ministérielle de bandes magnétiques avec le résultat des infections afin de pénaliser les hôpitaux non préoccupés par ce problème.

- * Comparer deux hôpitaux de même taille et de même profil serait intéressant dans le cadre des cliniques Sainte-Anne et Saint-Jean qui sont connectées au même système informatique.
- * Archivage de résultats à long terme à prévoir.
- * Etant responsable de l'application RCM (Resume Clinique Minimal), il me paraît particulièrement intéressant de confronter également les résultats laboratoires aux autres données médicales que notre système informatique possède.

Toutefois, pour élargir le spectre d'étude épidémiologique, une telle confrontation de données exigerait une révision de l'encodage de ces renseignements médicaux.

PARTIE 5 : BIBLIOGRAPHIE

1. JOHN V. BENNETT, M.D. and PHILIP S. BRACHMAN, M.D
Hospital infections
Edition Little, Brown and Company - Boston
2. CHASTAGNER Daniel
Hygiène hospitalière : un bon placement et un impératif économique
Gestion hospitalière, n° 230 pp 805-806
3. GORDTS B., MEYSMANS F.
Surveillance bactériologique à l'hôpital
Renseignements et directives
Publicité reçue par la clinique Saint-Jean de la firme Lilly Benelux
4. GORDTS B.
Toezicht op resistent kiemen in niet universitaire ziekenhuizen
Laboratorium Sint-Jan voor mikrobiologie, Bruges pp 17-19
5. HAXHE
Prévention des infections postopératoires des plaies
Hygiène hospitalière n° 2, 1984 pp 13-24
6. ROBERT W. HALLEY, DAVID H. CULKER, JOHN W. WHITE,
W. MEADE MORGAN, T. GRACE EMORI, VAN P. MUNN and
THOMAS M. HOOTON
The efficacy of infection surveillance and control programs
in prevention nosocomial infections in hospitals.
American journal of epidemiology, vol 121, n° 2 pp 182-205
7. HARTEMANN P.
Hygiéniste hospitalier : pour quoi faire ?
Gestion hospitalière, n° 250, nov 1985 pp 777-779
8. ICHARD
Infections nosocomiales : définitions - classification
Hygiène hospitalière n° 1 pp 9-12
9. INSTITUT D'HYGIENE ET D'EPIDEMIOLOGIE
Enquête de prévalence des infections nosocomiales, mars 1984
Rapport final 1985
10. INSTITUT D'HYGIENE ET D'EPIDEMIOLOGIE
Surveillance des maladies infectieuses par les laboratoires
de microbiologie en 1984
Rapport octobre 1985
11. M. JENNICK
Introduction à l'épidémiologie
Edition Malaise pp 358-389

12. MAISONNET, DAUBERT, VASSAL
Logiciel d'études épidémiologiques des infections hospitalières
Gestion hospitalière n° 244 mars 1985 pp 255-259
13. O.M.S.
Infections nosocomiales
Relevé épidémiologique hebdomadaire, 1985, pp 60, 281-288
14. C. RUMEAU, ROUQUETTE, G. BREART, R. PADIEU
Méthodes en épidémiologie
Collection statistiques en biologie et en médecine pp 2-19
15. WILLIAM E. SCHECKLER
Nosocomial infection control in the smaller hospital
What do we know, what do we do ?
Nosocomial infection - Yorke Medical Books, 1981
pp 14-17, 58-62
16. WILLIAM E. SCHECKLER
Hospital costs of nosocomial infections
A prospective three-month study
Infection control, 1980, pp 150-152
17. WILLIAM E. SCHECKLER
Nosocomial infection in a community hospital 1972-1976
Archives of Internal Medicine vol 138, dec 1978 pp 1792-1794
18. WILLIAM E. SCHECKLER, PATTY J. PETERSON, RN, MS, CIC
Nosocomial Infections in 15 rural Wisconsin Hospitals
Results and Conclusions from 6 Months of Comprehensive
Surveillance
Infection Control 1986, vol 7, pp 397-402
19. SUSAN H. TROXLER, RN, MPH
Infection Control Software for Microcomputers
Infection Control, 1986, vol 7, pp 470-472
20. VANDERVEIKEN F.
Surveillance des infections hospitalières
Hygiène hospitalière n° 2, 1983, pp 25-26
21. WAUTERS
Identifications des bactéries dans la surveillance des
infections hospitalières
Hygiène hospitalière, 1983, pp 27-28
22. Microinformatique et lutte contre les infections
Relevé épidémiologique hebdomadaire n° 12-23, mars 1984
23. BODART FR.
Éléments de conception et d'analyse des systèmes
d'information des organisations
Juillet 1981

- 24. VAN LAMSWEERDE A.
Cours d'analyse organique
- 25. HAINAUT J.L.
Cours de banques de données

FACULTES
UNIVERSITAIRES
N.D. DE LA PAIX

NAMUR



INSTITUT D'INFORMATIQUE

SURVEILLANCE EPIDEMIOLOGIQUE PAR
PROGRAMMES INFORMATISES A LA CLINIQUE SAINT-JEAN
DE BRUXELLES

Pascale LEFEBVRE

A N N E X E S

Memoire presente en vue de
l'obtention du titre de
Licencie et Maitre en
informatique.

Promoteur : J. FICHEFET

Annee academique : 1986-1987

A N N E X E S

I. ETUDE DE L'EXISTANT

1. Diagrammes des flux
2. Dynamique des traitements
3. Ecrans
4. Listes

II. ETUDE ET DESCRIPTION DE LA SOLUTION PROJETEE

1. Diagrammes des flux
2. Dynamique des traitements
3. Ecrans
4. Listes

III. ANALYSE FONCTIONNELLE

1. Description des donnees - schema entite/association
2. Description des traitements - ecrans et listes

IV. ANALYSE ORGANIQUE

1. Architecture des traitements
2. Schemas des acces necessaires

A N N E X E I.

1. Diagrammes des flux

1.1. Laboratoire

1.2. Admission

2. Dynamique des traitements

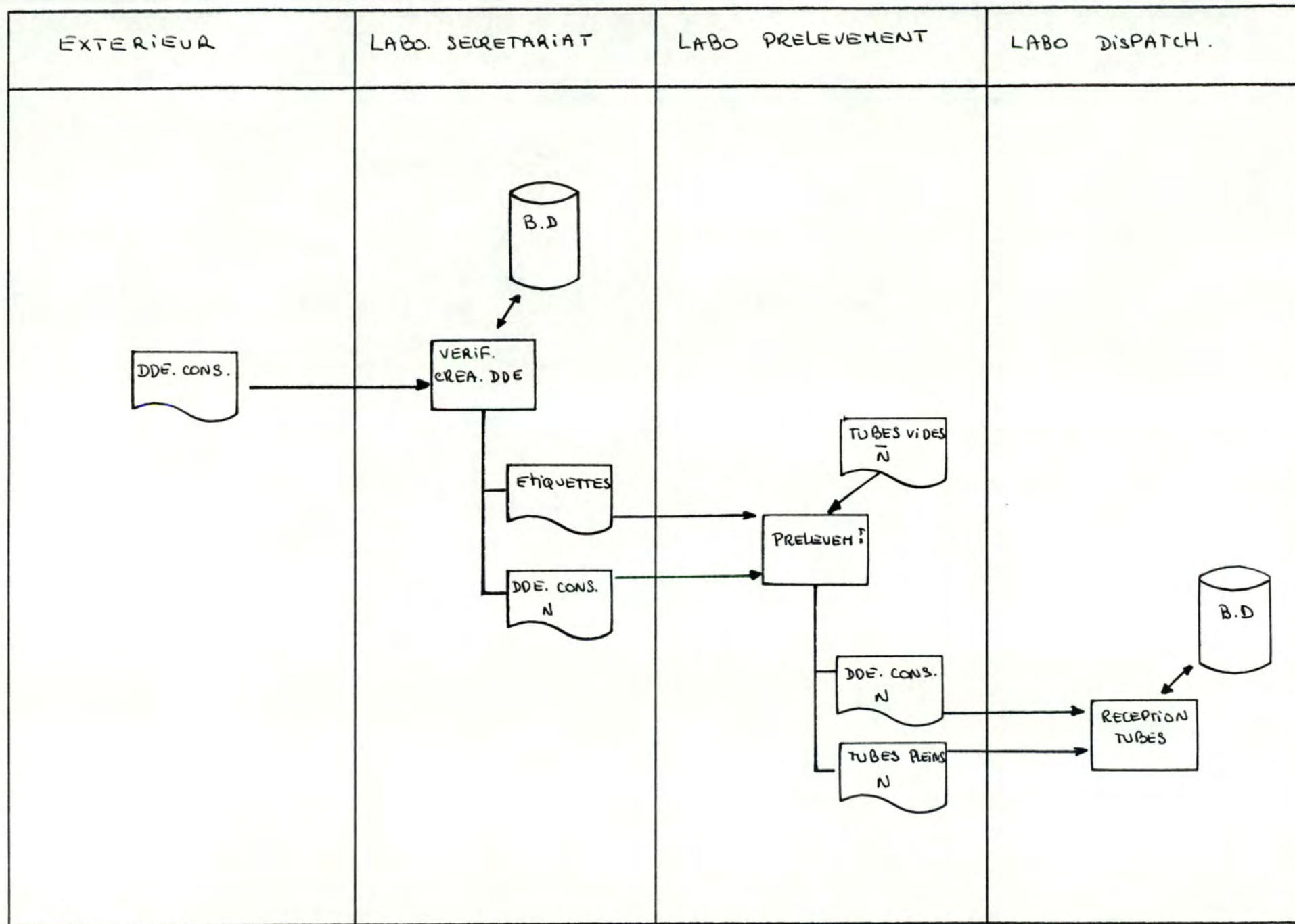
2.1. Encodage des resultats de bacteriologie

3. Ecrans

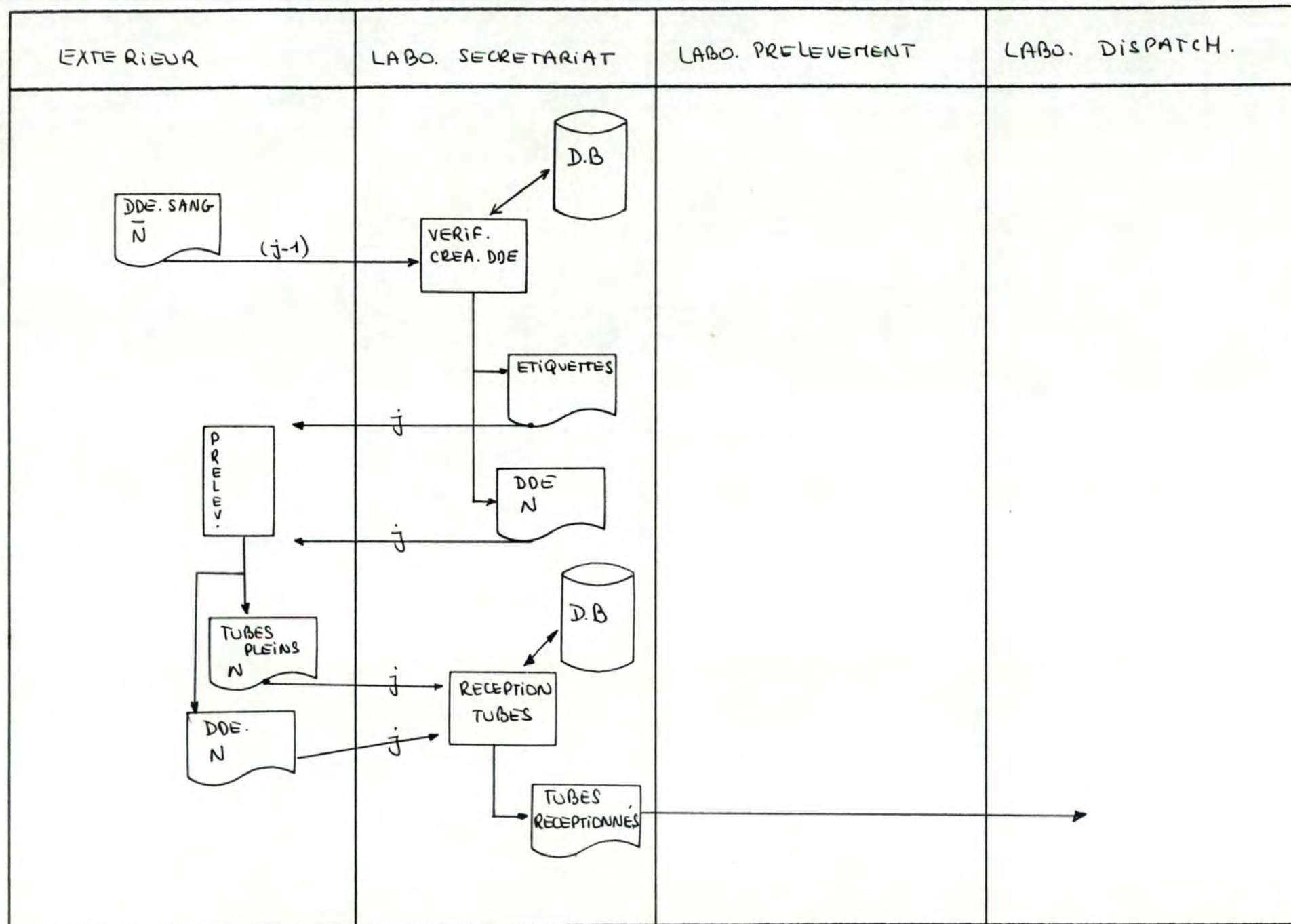
3.1. Encodage des resultats de bacteriologie

4. Listes

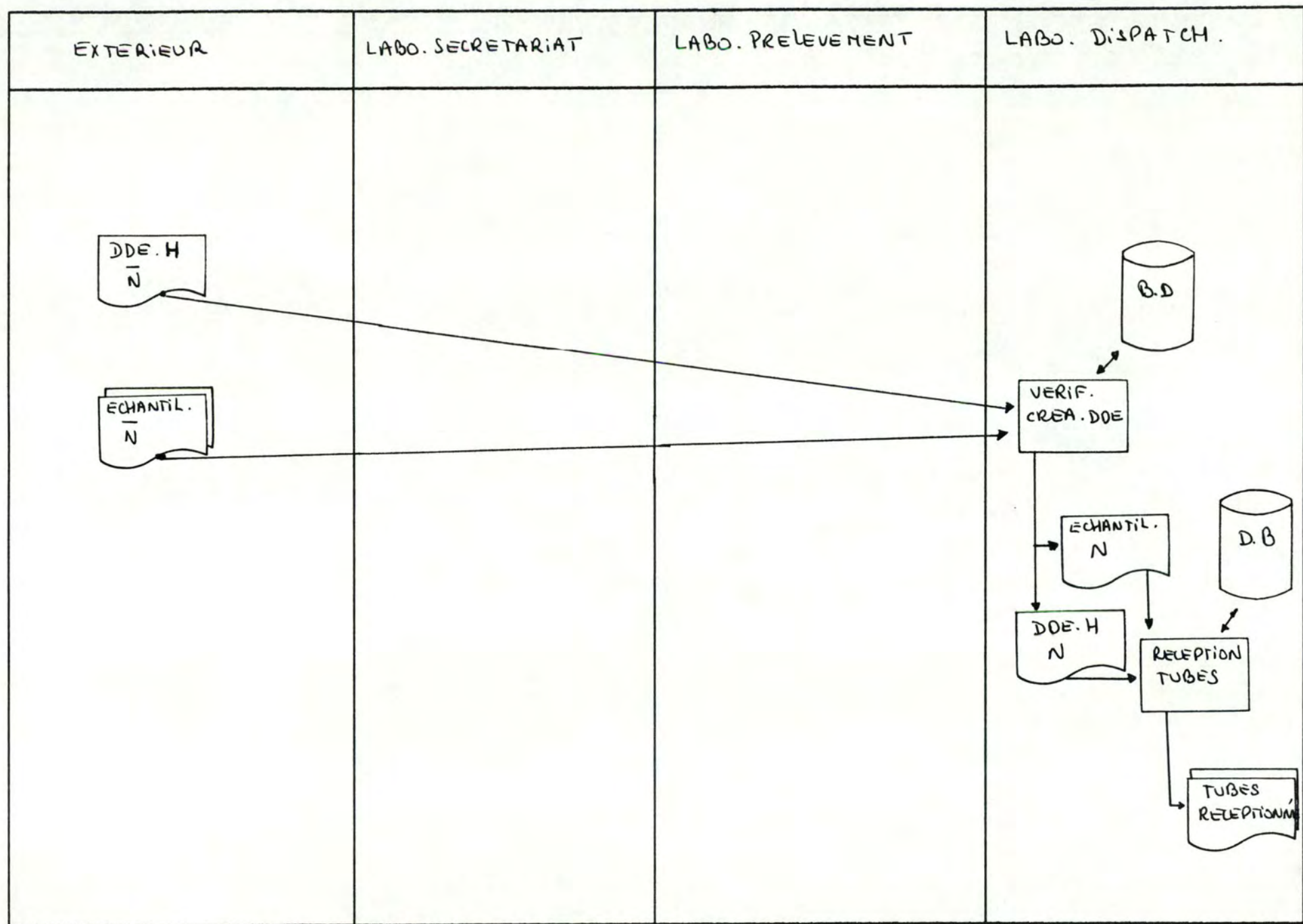
4.1. Protocoles de bacteriologie



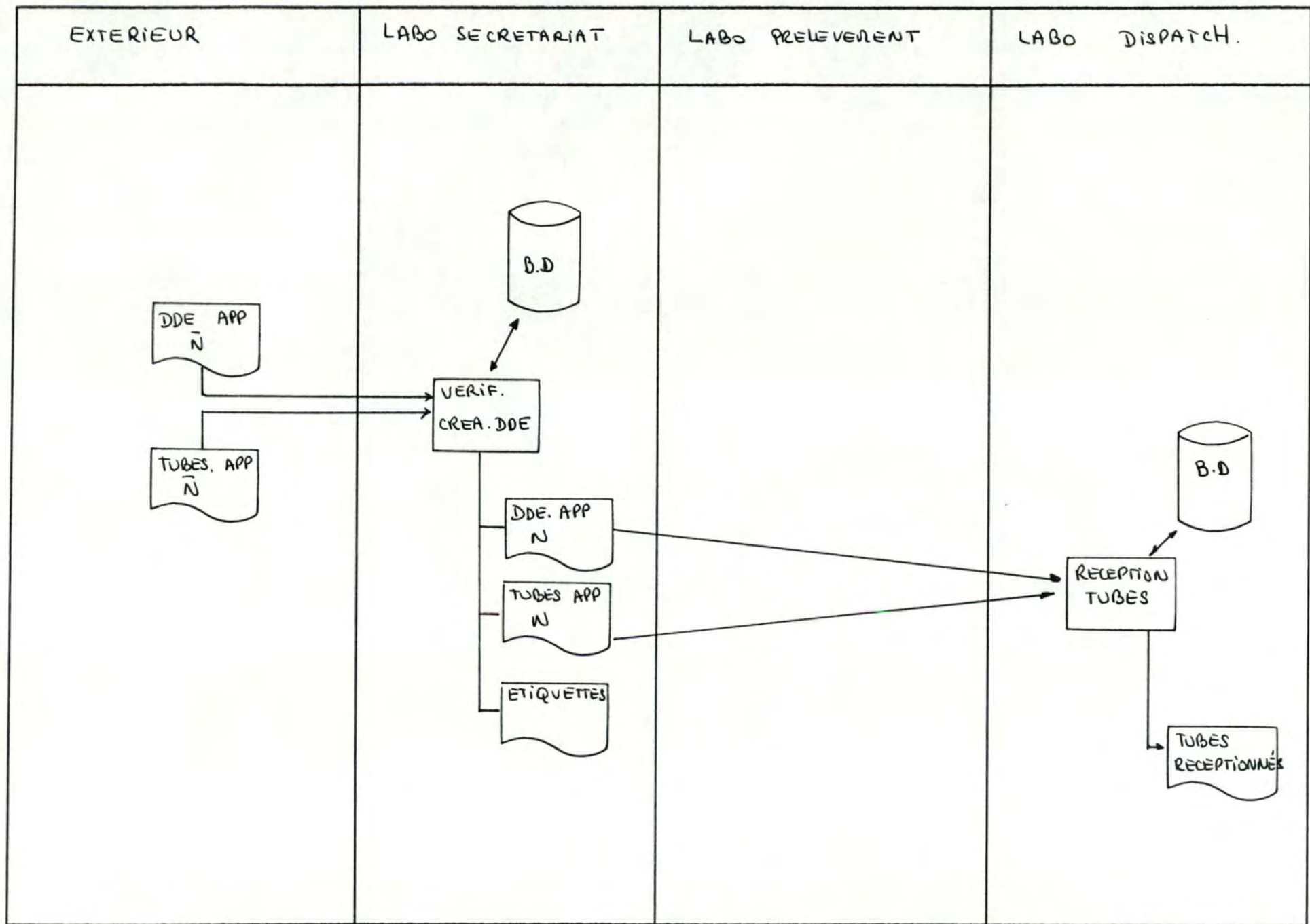
① DEMANDES "CONSULTANTS"



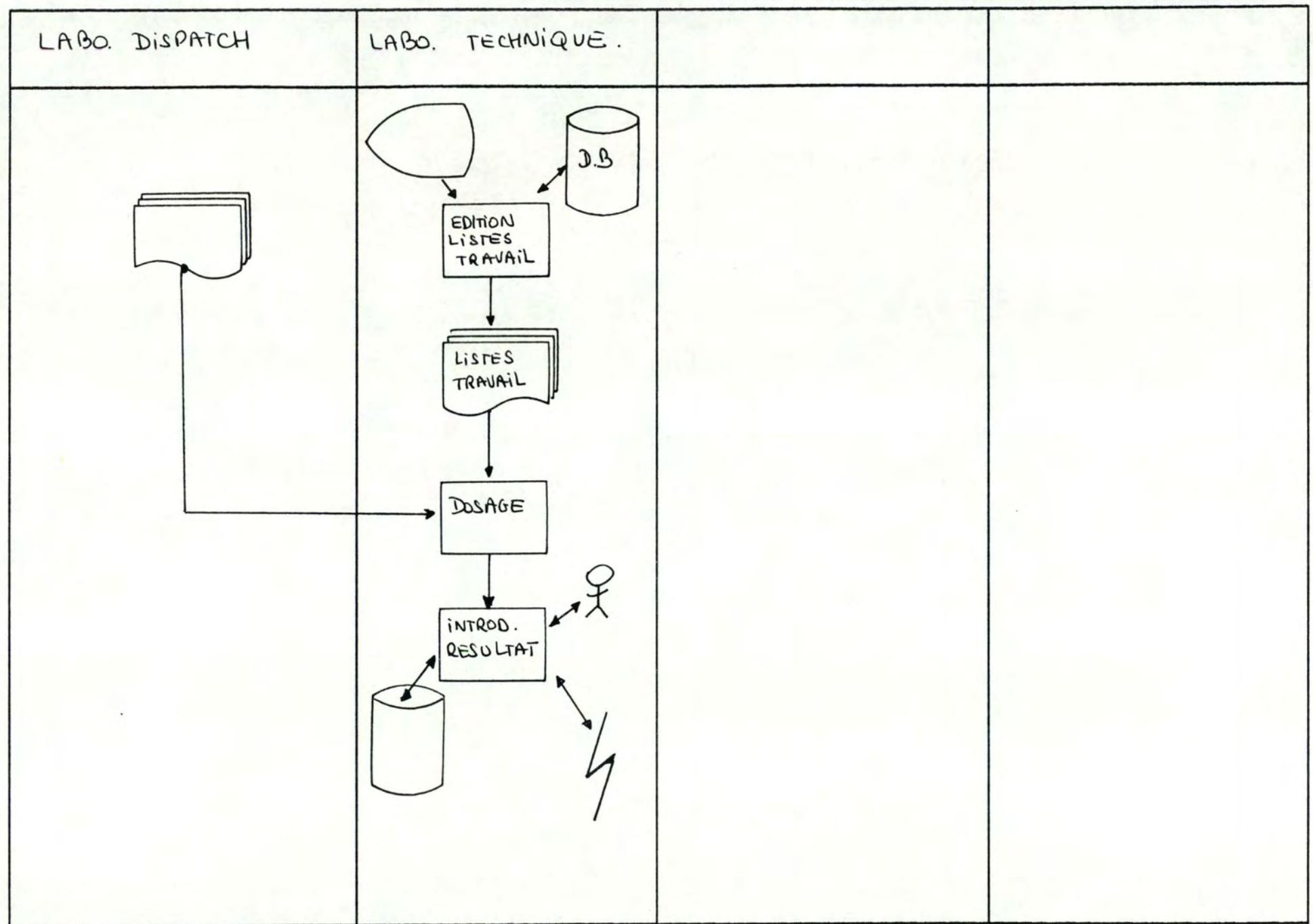
DEMANDES "HOSPITALISÉS NON URGENTES SUR SANG"



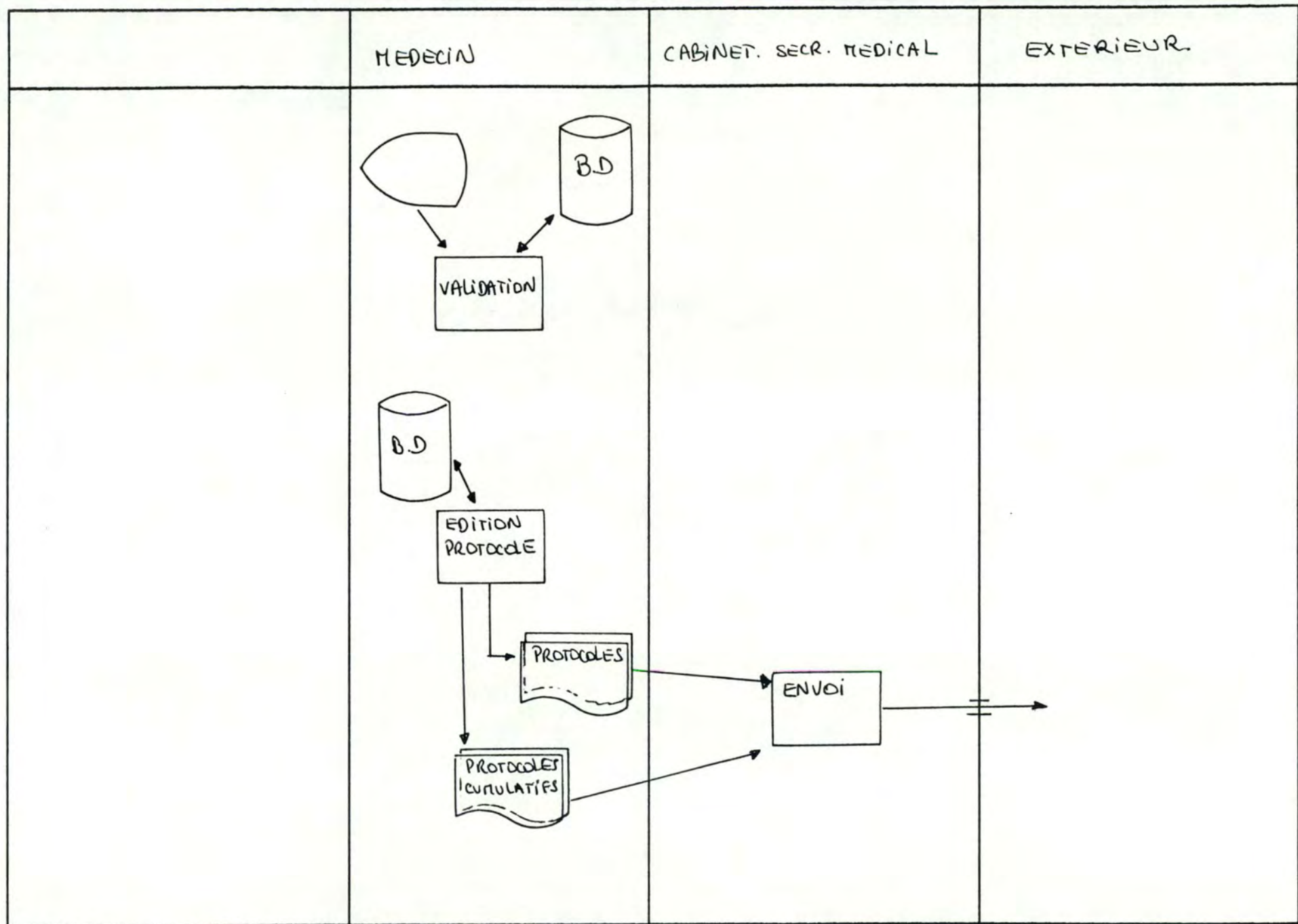
DEMANDES "HOSPITALISES NON URGENTS SUR LIQUIDES DIVERS".



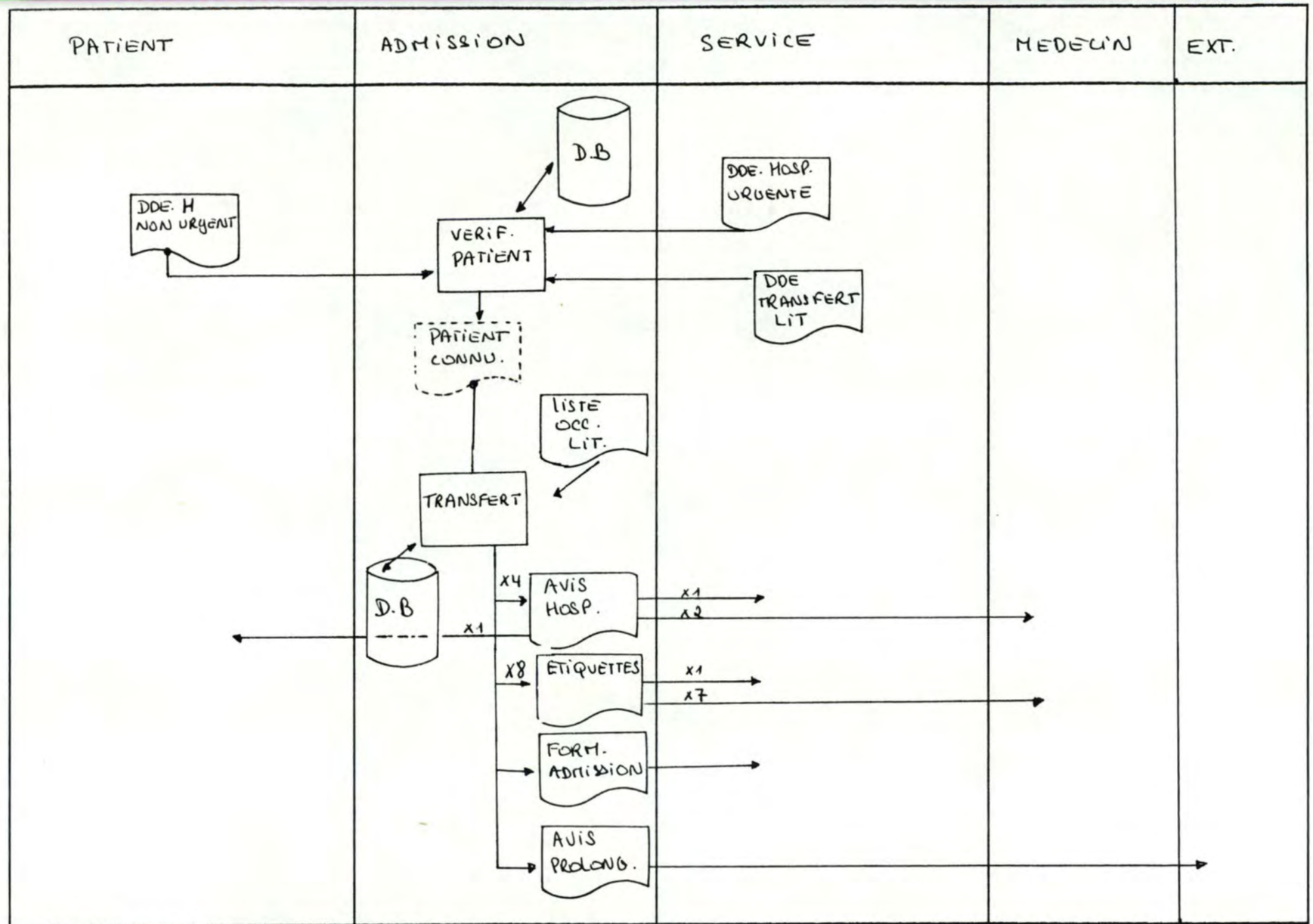
DEMANDES "APPORTEES" OU "HOSPITALISES URGENTS".



ENCODAGE DES RESULTATS.



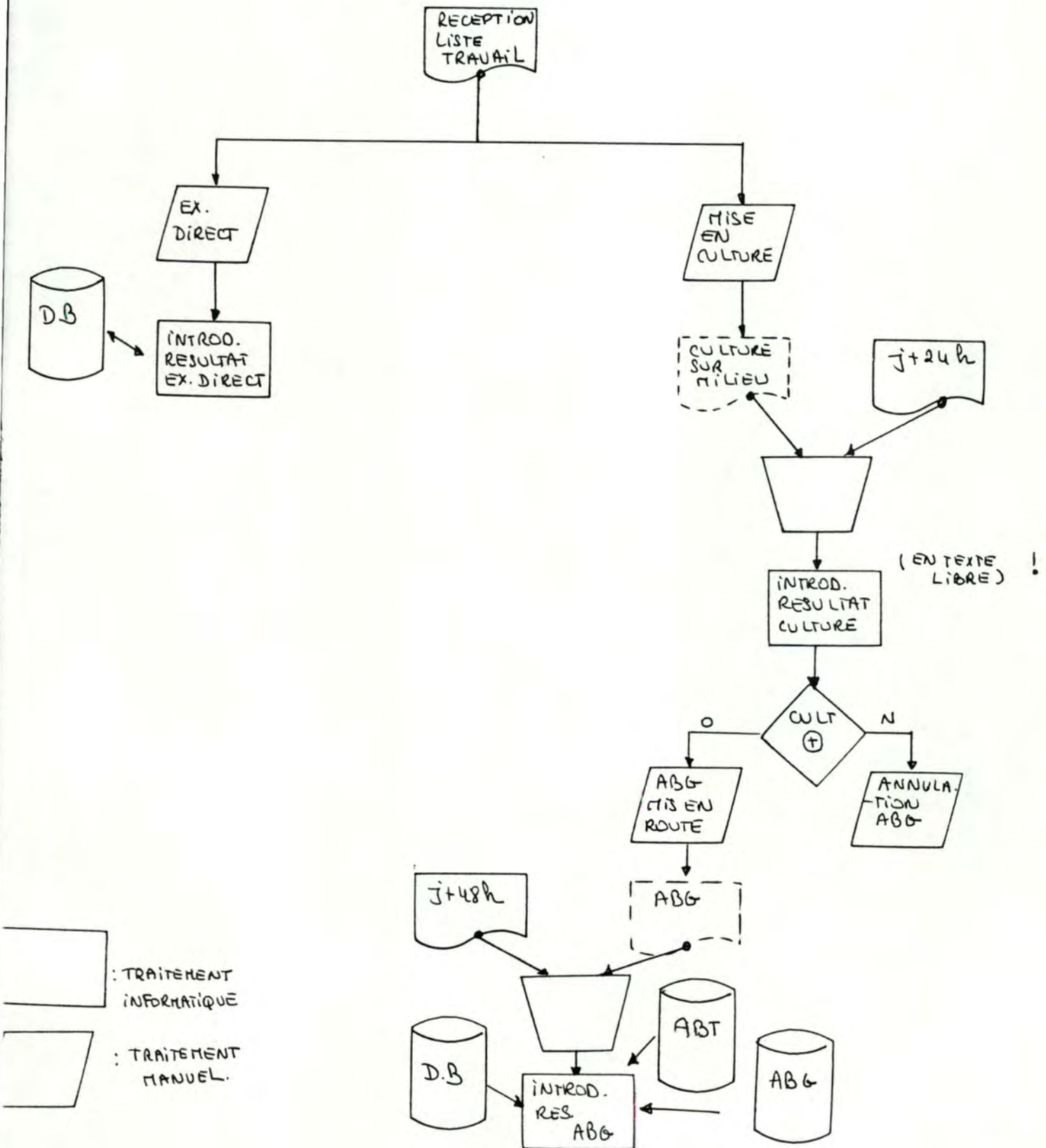
VALIDATION ET EDITION DES PROTOCOLES.



ADMISSION ET TRANSFERT DE LIT.

DYNAMIQUE TRAITEMENTS - ANCIENNE VERSION.

ENCODAGE DES RESULTATS DE BACTERIOLOGIE.



Ex.direct ur.

Cult.aerobe ur.

Autres codes analyses ?

EDUR ? [@NNN @2B @AL @QQ @2BB]
? []

EDUR tres nombreux bacilles gram negatif
 quelques granulocytes neutrophiles

Correction (O/N) : N ?

C.UR ? [@2U5 @AL @2S1 @2343 @AL @2S2 @2343 @AL @2S3 @2521]
? []

C.UR plus de cent mille germes par ml :
 souche un : KLEBSIELLA PNEUMONIAE.
 souche deux : KLEBSIELLA PNEUMONIAE.
 souche trois : PSEUDOMONAS AERUGINOSA.

Correction (O/N) : N ?

ok ?

C.UR; analyse terminee (O/N) : N ?

ECRAN : ENCODAGE DES RESULTATS DE BACTERIOLOGIE.

Analyses Normes/Unites

26-08-1987
18:22

BACTERIOLOGIE

Ex.direct ur.	"	tres nombreux bacilles gram negatif quelques granulocytes neutrophiles tres nombreux bacilles gram negatif quelques granulocytes neutrophiles
Cult.aerobe ur.	"	plus de cent mille germes par ml : souche un : KLEBSIELLA PNEUMONIAE. souche deux : KLEBSIELLA PNEUMONIAE. souche trois : PSEUDOMONAS AERUGINOSA.
ABg aero.urines		N.recept
??		
??		
??		
??		
??		

Protocole de BACTERIOLOGIE.

A N N E X E II.

1. Diagrammes des flux

- 1.1. Creation/Maj table
- 1.2. Edition table
- 1.3. Traitements journaliers
- 1.4. Listes des patients infectes
- 1.5. Autres statistiques

2. Dynamique des traitements

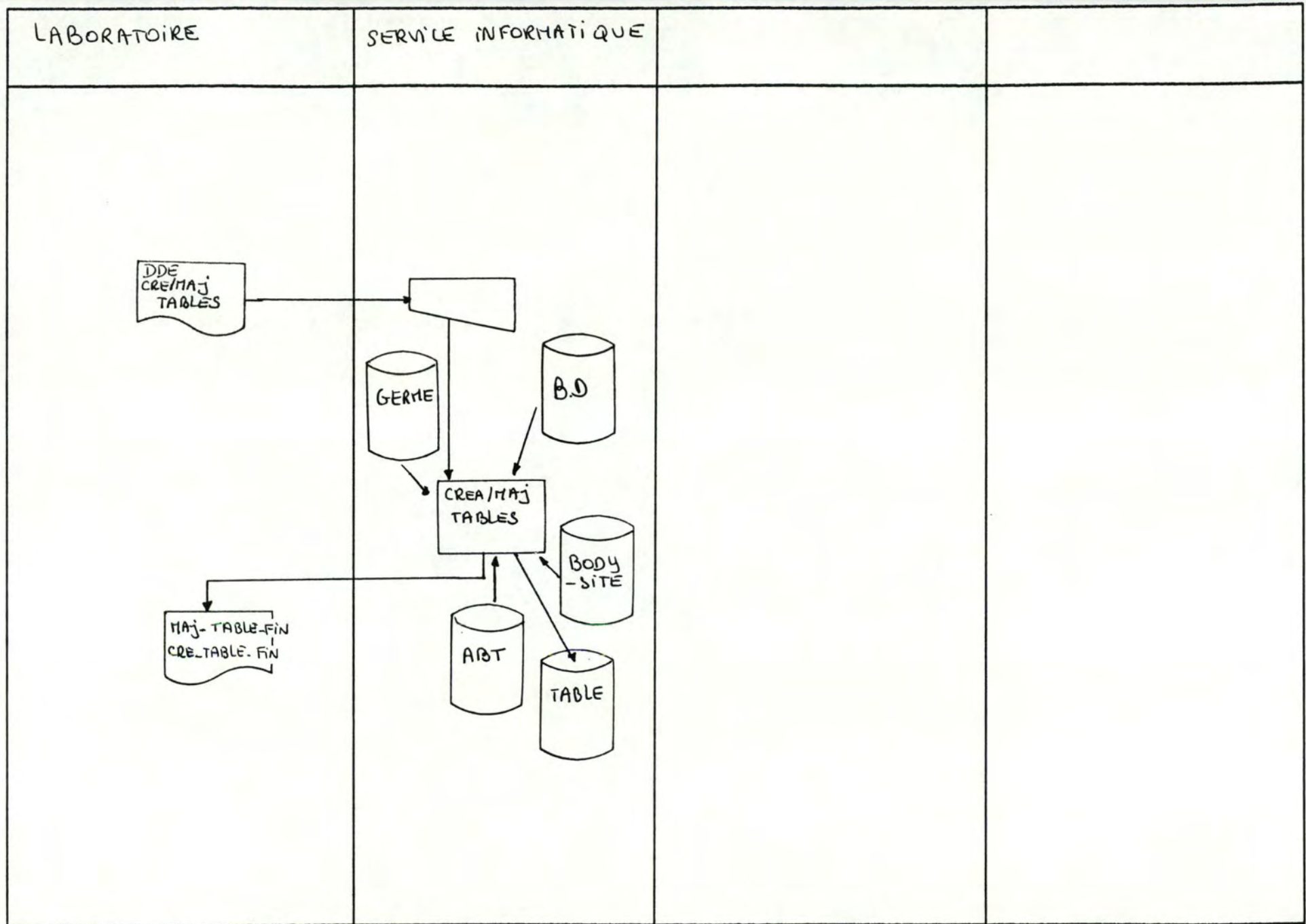
- 2.1. Encodage des resultats

3. Ecrans

- 3.1. Encodage des demandes de bacteriologie
- 3.2. Encodage des resultats de bacteriologie

4. Listes

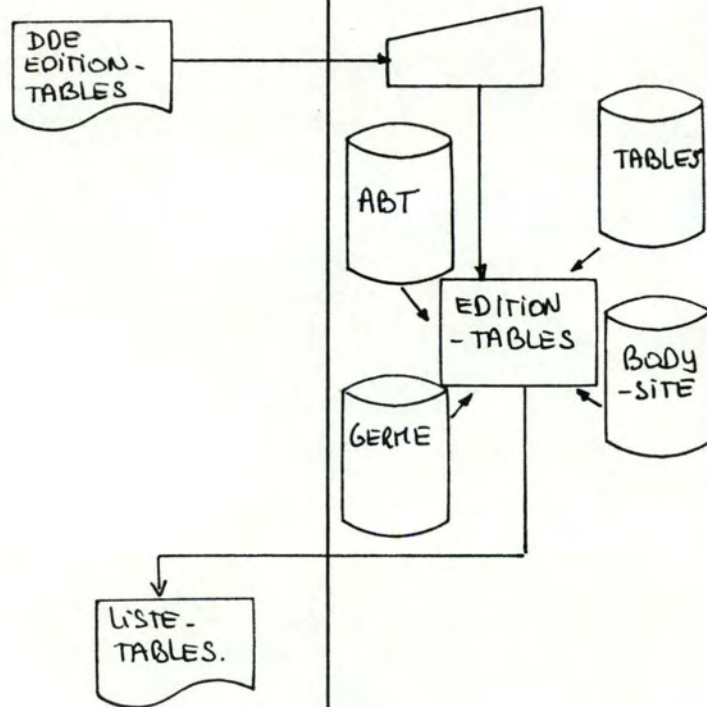
- 4.1. Protocoles de bacteriologie



CREA/MAJ - TABLES.

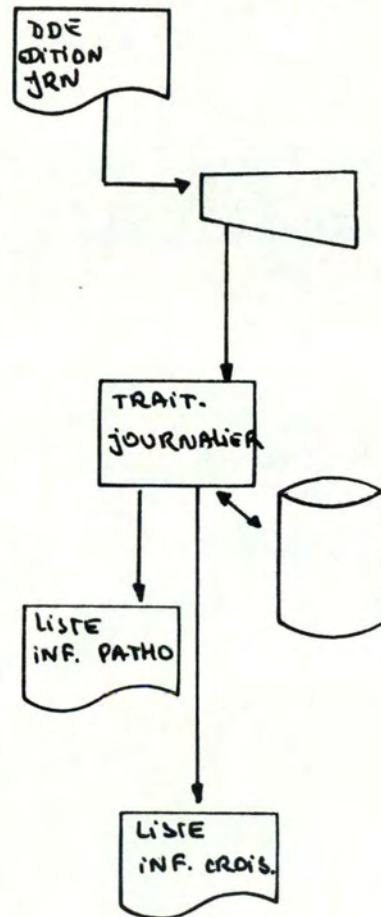
LABORATOIRE

SERVICE INFORMATIQUE

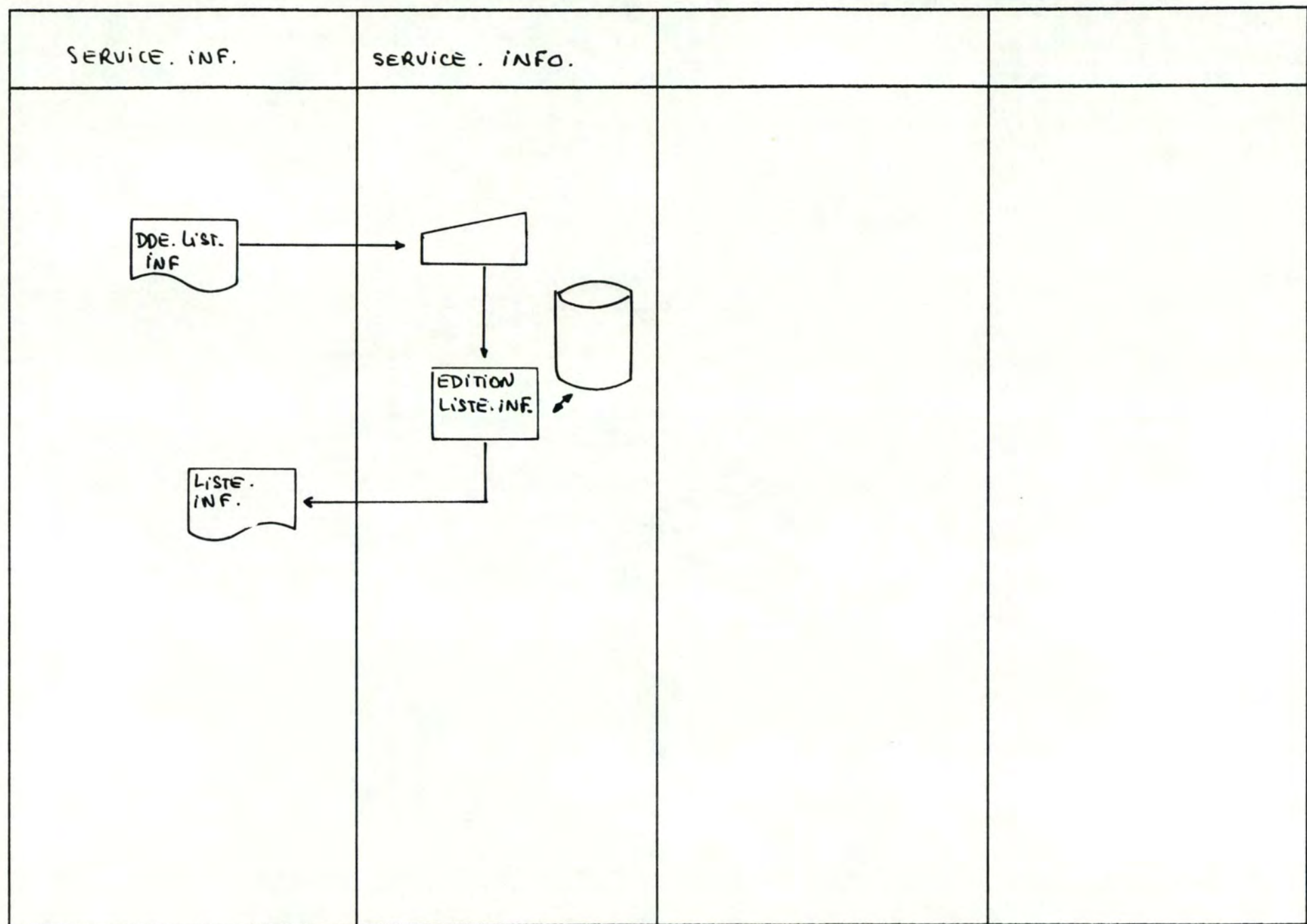


EDITION - TABLES.

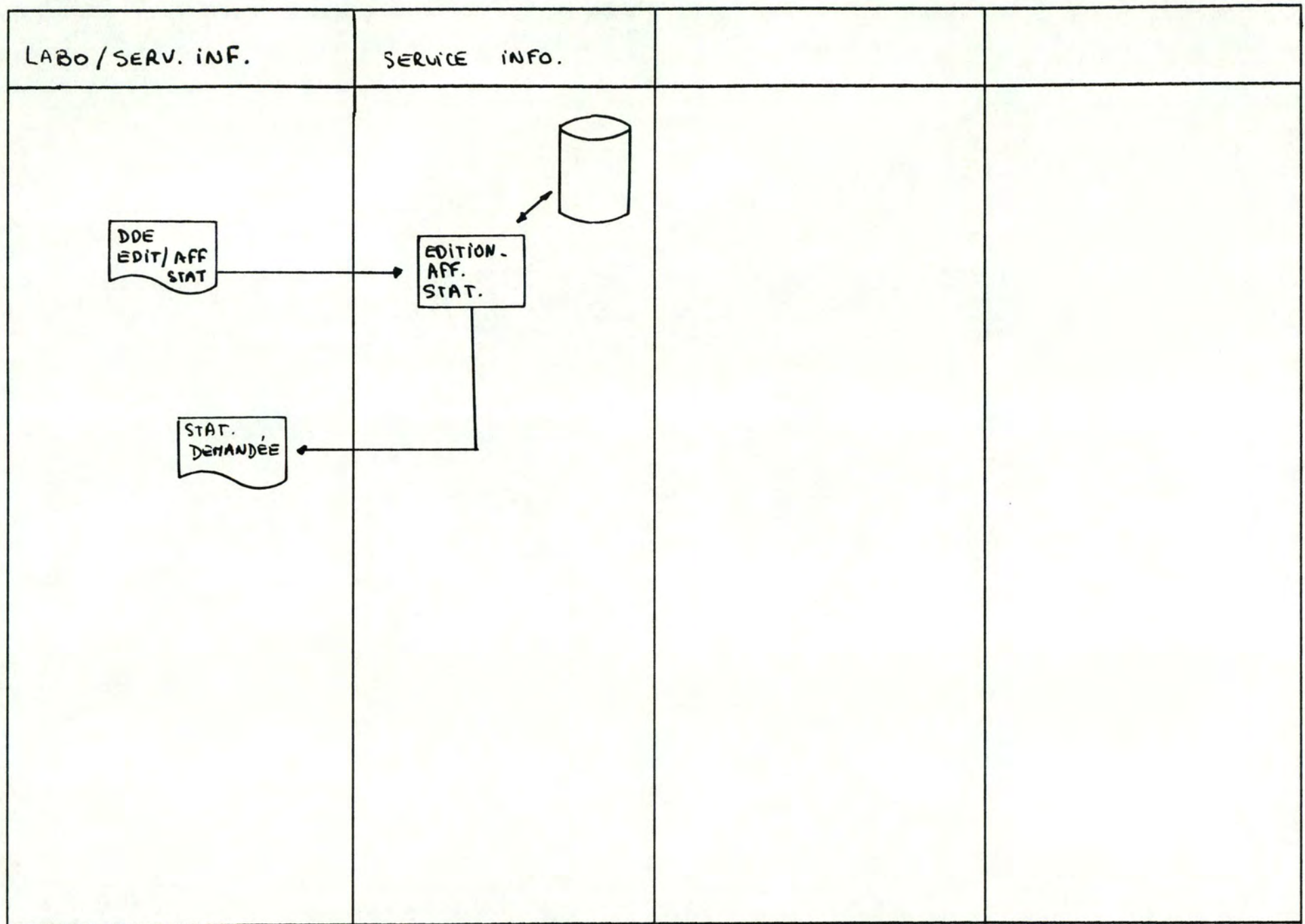
SERVICE INFORMATIQUE.



TRAITEMENTS JOURNALIERS.



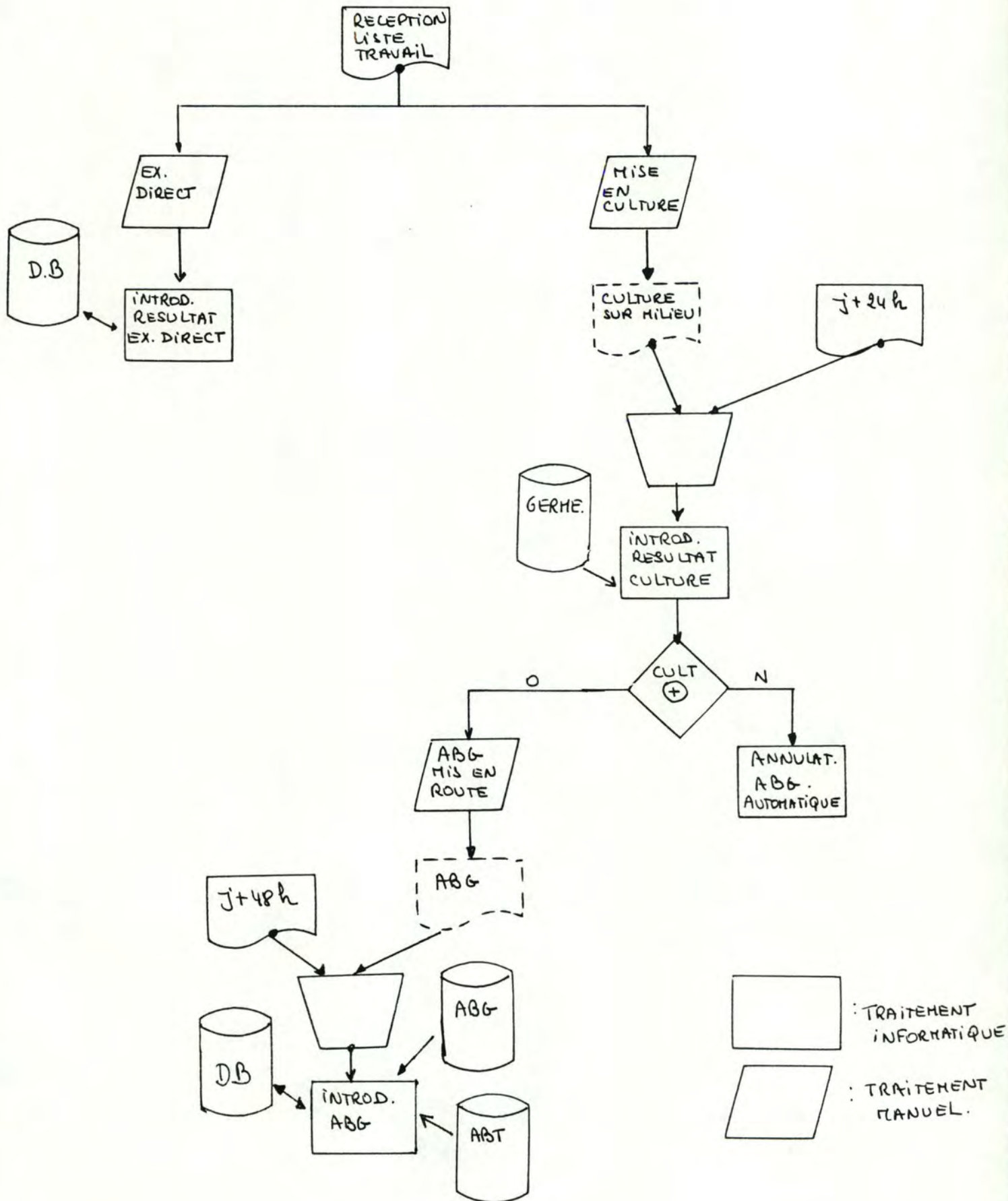
EDITION LISTES INFECTIONS TRIÉES .



EDITION / AFFICHAGE STATISTIQUES.

DYNAMIQUE TRAITEMENTS . NOUVELLE SOLUTION.

ENCODAGE DES RESULTATS DE BACTERIOLOGIE.



ENCODAGE DE LA DEMANDE BACTERIOLOGIE.

Ok (Numero/*/O/N) : 0 ?

Demande attribuee : 01'00095

01'00095 [REDACTED] CHRISTIANE 150149 F 3000009; 12 24-07-87 21:15, J106.1/1

Code : ORPU

"Body-site" : PU , Pus ? *

[1] S/F pus

[2] S/F pus 2

Numero ? 1

[1] PUG , Pus gyneco.

[2] FUGU , Pus gyneco.urethral

[3] FUGUG , Pus gyneco.uretral gche

[4] FUGV , Pus gyneco.vaginal

Numero ? 2

Reception terminee : 01'00095

??

??

??

INTRODUCTION DES RESULTATS "BODY-SITE"

58'18648 [REDACTED] GERMAINE 050412 F 6289590; 12 09-09-87 20:30, J659.

Code : DRUR

(BS) "B-S"(/, #, -) : U , Urines ?

ENCODAGE DES RESULTATS DE BACTERIOLOGIE.

INTRODUCTION DES RESULTATS DES GERMES

58'17811 ██████████ SUZANNA 260314 F 3290158; 7757 08-09-87 10:36, M510.
Code : C.UR

(CL) Culture	:	P ?		
Negatif	:		?	
Positif	:		?	
Info.germe	:			?
(NG) Num. gen.	:		?	
(SH) Souche	:	1 ?		
Numeration	:	Plus de 100.000 germes/ml	?	
Germe	:	SELI , Serratia liquefaciens		?

(ID) Acc. identif. (O/N) : ?
Culture Positive : P

ENCODAGE DES RESULTATS DE BACTERIOLOGIE.

IDENTIFICATION DE L'ANTIBIOGRAMME

58'17811 ██████████ SUZANNA 260314 F 3290158; 7757 08-09-87 10:36, M510.

Souche 2 : BNID , bacilles gram (-) a identifier

(TP) Type antibiogramme 2 ? (SP) Antibiotiq. sup. ?
Resultat ABt. sup. ?

Souche(s) : 1 2 1 2

(1)	1	AMO	:	R	R	?	(8)	8	NIT	:	R	R	?
(2)	2	PIP	:	R	S	?	(9)	9	NOR	:	R	S	?
(3)	3	CEF	:	R	R	?	(10)						?
(4)	4	CET	:	S	I	?	(11)						?
(5)	5	TOB	:	R	S	?	(12)						?
(6)	6	AMI	:	S	S	?	(13)						?
(7)	7	SXT	:	R	S	?	(14)						?

Ok (Rubrique/*/O/N) : 0 ?

ENCODAGE DES RESULTATS DE BACTERIOLOGIE.

BACTERIOLOGIE

Urines
Origine urines
Ex.direct ur.

Urines
nombreux bacilles gram negatif
rares coques gram positif
rares bacilles gram positif
rares granulocytes neutrophiles

Cult.aerobe ur.

Positive

Souche : 1

Numeration : Plus de 100.000 germes/ml
Germe : Serratia liquefaciens

Souche : 2

Numeration : Plus de 100.000 germes/ml
Germe : bacilles gram (-) a identifier

ABg aero.urines

Souches : 1 2

Amoxicilline	R	R
Piperacilline	R	S
Cefazoline	R	R
Cefotaxime	S	I
Tobramycine	R	S
Amikacine	S	S

NOUVEAU PROTOCOLE DE BACTERIOLOGIE.

A N N E X E III.

1. Description des donnees

1.1. Schema entite/association

2. Description des traitements

2.1. Ecrans

- 2.1.1. Acceptation commandes/saisie-cde
- 2.1.2. Acceptation commandes/sel-body
- 2.1.3. Creation/Maj germe
- 2.1.4. Creation/Maj body-site
- 2.1.5. Creation/Maj generalites body-site
- 2.1.6. Creation/Maj particularites germe body-site
- 2.1.7. Creation/Maj germe-cible
- 2.1.8. Creation/Maj germe-pathogene
- 2.1.9. Acceptation commandes epidemiologie/saisie-commandes-epi
- 2.1.10. Acceptation commandes epidemiologie/saisie-rubrique-list-inf-epi
- 2.1.11. Acceptation commandes epidemiologie/saisie-rubrique-stat-epi

2.2. Listes

- 2.2.1. Edition germe
- 2.2.2. Edition body-site
- 2.2.3. Edition listes trieess/list-inf-1
- 2.2.4. Edition listes trieess/list-inf-2
- 2.2.5. Edition statistiques/stat-1/repartition des germes par service
- 2.2.6. Edition statistiques/stat-2/taux d'infection par service
- 2.2.7. Edition statistiques/stat-3/taux d'infection par body-site et par service
- 2.2.8. Edition statistiques/stat-4/histogramme de sensibilite des germes
- 2.2.9. Edition statistiques/stat-5/polyresistance des germes
- 2.2.10. Traitement journalier/liste pathogenes
- 2.2.11. Traitement journalier/liste cibles

EDITION SUR ECRAN

?? MUT

- [1] Gestion des antibiotiques
- [2] Gestion des antibiogrammes
- [3] Test RAST
- [4] Gestion des mixtures
- [5] Gestion des body-sites
- [6] Gestion des germes

Numero ?

-----< GESTION DES FICHIERS ANNEXES -----

EDITION SUR ECRAN

?? MUT 5

- [1] Gestion body-sites
- [2] Gestion Master/Slave
- [3] Gestion generalites body-site
- [4] Gestion germes pathogenes
- [5] Gestion germes cibles

Numero ?

-----< LABO - GESTION DES GERMES >-----

(CG) Code germe :

(GG) Genre de germe :

(EG) Espece de germe :

(SG) Serotype :

(FG) Forme :

(CO) Coloration :

(LF) Libelle fr. :

(LN) Libelle nl. :

Modifier, Supprimer, Quitter :

-----< LABO - GESTION DES BODY-SITES >-----

(CB) Code body-site :

(LF) Libelle fr.body-site :

(LN) Libelle nl.body-site :

(SF) Libelle fr.sous-famille :

(SN) Libelle nl.sous-famille :

(ML) Def. numeration (G=1,S=2,*) :

(XL) 1/2 :

(MT) Numeration min. :

(XT) max:

Modifier, Supprimer, Quitter :

GESTION DES GENERALITES "BODY-SITES/GERMES"

- (CB) Code body-site ? P : Pus (non precise)
- (CG) Code germe ? STAU : Staphylococcus aureus
- (TN) Type de numeration (qualitatif=QL,quantitatif=QT) : QL
- (NM) Numeration : 16- 30 ? 2
- (DA) Duree non redondance : 0 ? 7
- (DB) Duree recherche liste p. : 0 ? 30
- (CA) Code antibiogram. 2 : ? B

Mode creation de nouvelles donnees

GESTION DES GERMES "CIBLES"

(CB) Code body-site ? P : Pus (non precise)
(CG) Code germe ? PSAE : Pseudomonas aeruginosa
(TN) Type de numération (qualitatif=QL,quantitatif=QT) : QL
(NM) Numération : 0- 32767 ?
Duree non redondance "F" : 7
(DR) Duree recherche cand. C. : 45 ?
(CA) Code antibiotique 1 : ?
(RS) Resultat a tester (S/I/R) : . ?

Mode mise a jour des donnees
Ok (Rubrique/*/O/N) : 0 ?

GESTION DE LA PATHOGENICITE DES GERMES

(CB) Code body-site ? P : Pus (non precise)
(CG) Code germe ? STAU : Staphylococcus aureus
(TN) Type de numération (qualitatif=QL,quantitatif=QT) : QL
(NM) Numération : 0- 32767 ?
Duree non redondance "P" : 7
(DR) Duree recherche liste p. : 30 ?
(CA) Code antibiotique 3 : ?
(RS) Resultat a tester (S/I/R) : . ?

Mode mise a jour des donnees
Ok (Rubrique/*/O/N) : 0 ?

-----< EPIDEMIOLOGIE : LISTES ET STATISTIQUES >-----

TERMINAL : 10

?? SOS

SOS HELP KB TERM FRAN NEDE LIST STOP BYE

??

-----< EPIDEMIOLOGIE : LISTES ET STATISTIQUES >-----

TERMINAL : 10

?? LIST

- [1] Liste des infections journalieres (pathologiques/croisees)
- [2] Listes diverses des patients infectes
- [3] Statistiques epidemiologiques

Numero ?

Listes des patients infectes

- [1] Liste des patients par GERME/SERVICE
- [2] Liste des patients par SERVICE/GERMES

Numero ?

Statistiques epidemiologiques

- [1] Repartition des germes par service
- [2] Taux d'infection par service
- [3] Taux d'infection par body-site et par service
- [4] Histogramme de sensibilite
- [5] Germes polyresistants

Numero ?

DONNEES GENERALES DES GERMES

Lecture, tri et edition du fichier

ACAN	Acinetobacter anitratus	
ACLW	Acinetobacter lwoffii	
AEVI	Aerococcus viridans	
AGRA	Agrobacterium radiobacter	
AHXY	Achromobacter xylosoxidans	
ALDE	Alcaligenes denitrificans	
ALFA	Alcaligenes faecalis	
ALOD	Alcaligenes odorans	
ANA	Absence d'anaerobies stricts	Non identifie
AOHY	Aeromonas hydrophila	
ARHA	Arcanobacterium haemolyticum	
ASFU	Aspergillus fumigatus	
ASNI	Aspergillus niger	
ATIS	Actinomyces israeli	
ATOD	Actinomyces odontolyticus	
BAAS	Bacteroides asaccharolyticus	
BABI	Bacteroides bivius	
BACO	Bacteroides corrodens	
BADI	Bacteroides distasonis	
BADS	Bacteroides disiens	
BAEG	Bacteroides eggerthii	

DONNEES GENERALES DES BODY-SITES

Lecture, tri et edition du fichier

A	Aspiration non precisee	Aspiration	Master : E	S	QL
AB	Aspiration bronchique	Aspiration	Master : E	S	QL
AL	Liqu.lavage bronchoalv.	Expecto	Master : E	S	QL
AT	Aspiration tracheale	Aspiration	Master : E	S	QL
AX	Aspiration (a preciser)	Aspiration	Master : E	S	QL
B	Biopsie antrale			S	QL
BB	Biopsie bulbaire		Master : B	S	QL
BO	Biopsie oesophagienne		Master : B	S	QL
BX	Biopsie ... (a preciser)		Master : B	S	QL
C	Liquide cephalorachidien			S	QL
CV	Liquide ventriculaire		Master : C	S	QL
CX	Liquide a preciser		Master : C	S	QL
DB	Bile	Divers	Master : P	S	QL
DH	Hematome	Divers	Master : P	S	QL
DL	Lait maternel	Divers	Master : P	S	QL
DS	Squames	Divers	Master : P	S	QL
DV	Vomissements	Divers	Master : P	S	QL
DX	Divers (a preciser)	Divers	Master : P	S	QL
E	Expectorations	Expecto		S	QL
FA	Liquide amniotique	Gyneco	Master : P	S	QL
FB	Glande de Bartholin	Gyneco	Master : P	S	QL

Liste nominative des infectionsGerme : Haemophilus haemolytica Service : REA 01-01-87/30-03-87
=====

████████ Dominique 30-07-61 M 145577 REA
preleve le : 01-02-87 Labo : 1' 15 HAAA Pus(non precise)

████████ Pascale 19-11-59 F 458747 REA
preleve le : 15-02-87 Labo : 1' 57 HAAA Pus ulcere

████████ Christian 15-01-49 M 3000009 REA
preleve le : 22-03-87 Labo : 1' 67 HAAA Pus ulcere

Fin d'edition.

Repartition des germes par serviceGerme : Haemophilus haemolytica Service : * 01-01-87/30-03-87
=====

	Nbre	Nbre/Total	
Haemophilus haemolytica :	20	40%	MAT

	Nbre	Nbre/Total	
Haemophilus haemolytica :	30	60%	NEO

Total selectionne	: 50	
-------------------	------	--

Fin d'edition.

Taux d'infection par service

Service : REA

01-02-87/01-08-87

=====

	Nb prel	Nb inf	Nb inf/Nb prel	Nb inf/Nb T prel
Catheter	: 100	60	60%	35%
Gorge	: 45	15	33%	7.5%
Nez	: 65	50	71%	25%

Total selec.	: 200	125		62.5%

Fin d'edition

Taux d'infection par body-site et par service

Body-site : Catheter service: REA

01-02-87/01-08-87
=====

REA

	Nb prel	Nb inf	Nb inf/Nb prel	Nb inf/Nb T prel
Catheter	: 100			
Enterocoques	:	10	10%	10%
Pseudo. aerug.	:	30	30%	30%
Prot. vulg.	:	10	10%	10%

Total selec.	: 100	50		50%

Fin d'edition

Histogramme de sensibilite d'un germe

Germe : Haemophilus haemolytica

01-01-87/30-03-87

=====

	St	Nb	OXA	PEN	CEF	CFZ	CHLO	AMP	MYC
H.haemolytica	A	231	0%	99%	1%	10%	21%	9%	33%
	H	126	1%	98%	2%	10%	18%	5%	32%

Fin d'edition.

Polyresistance aux antibiotiques01-01-87/30-03-87
=====

Antibiotiques testés:	PEN	CEF	CFZ	CHLO	AMP	MYC
	Tous testés		tous resis.		%	
Haemophilus haemolytica :	50		20		40	
Pseudomonas aeruginosa :	20		1		5	

Fin d'édition.

Infections pathologiques Service lors du prelevement : 32

4019335 [REDACTED] EULAHIE 08-06-1901 F
Hospitalise le :14-07-1987 Numero ADM : 9670 Chambre/lit : J470.6/1 Dernier transf.:11-09-1987
Preleve le :14-09-1987 Labo : 1' 8 Pus (non precise) Haemophilus haemolyticus

6014562 [REDACTED] BRIGITTE 20-08-1958 F
Hospitalise le :23-07-1987 Numero ADM : 9673 Chambre/lit : J464.1/1 Dernier transf.:23-07-1987
Preleve le :14-09-1987 Labo : 1' 10 Pus (non precise) Haemophilus haemolyticus

8019336 [REDACTED] DELPHINE 01-04-1900 F
Hospitalise le :24-07-1987 Numero ADM : 9674 Chambre/lit : J586.1/1 Dernier transf.:24-07-1987
Preleve le :14-09-1987 Labo : 1' 9 Pus (non precise) Haemophilus haemolyticus

Candidats cibles Service lors du prelevement : 32
=====

8019336 [REDACTED] 01-04-1900 F
Hospitalise le :24-07-1987 Numero ADM : 9674 Chambre/lit : J586.1/1 Dernier transf.:24-07-1987
Preleve le :14-09-1987 Labo : 1' 9 Pus (non precise) Haemophilus haemolyticus

4019335 [REDACTED] 08-06-1901 F
Preleve le :12-09-1987 Numero ADM : 9670 Chambre/lit : 470.6
Labo : 1' 8 Edt.

3000009 [REDACTED] 15-01-1949 M
Preleve le :08-09-1987 Numero ADM : 9678 Chambre/lit : 106.1
Labo : 1' 7

Fin d'edition.

A N N E X E IV.

1. Architecture des traitements

1.1. Exemples de decoupe en modules

1.1.1. Creation/Maj germes

1.1.2. Traitement journalier

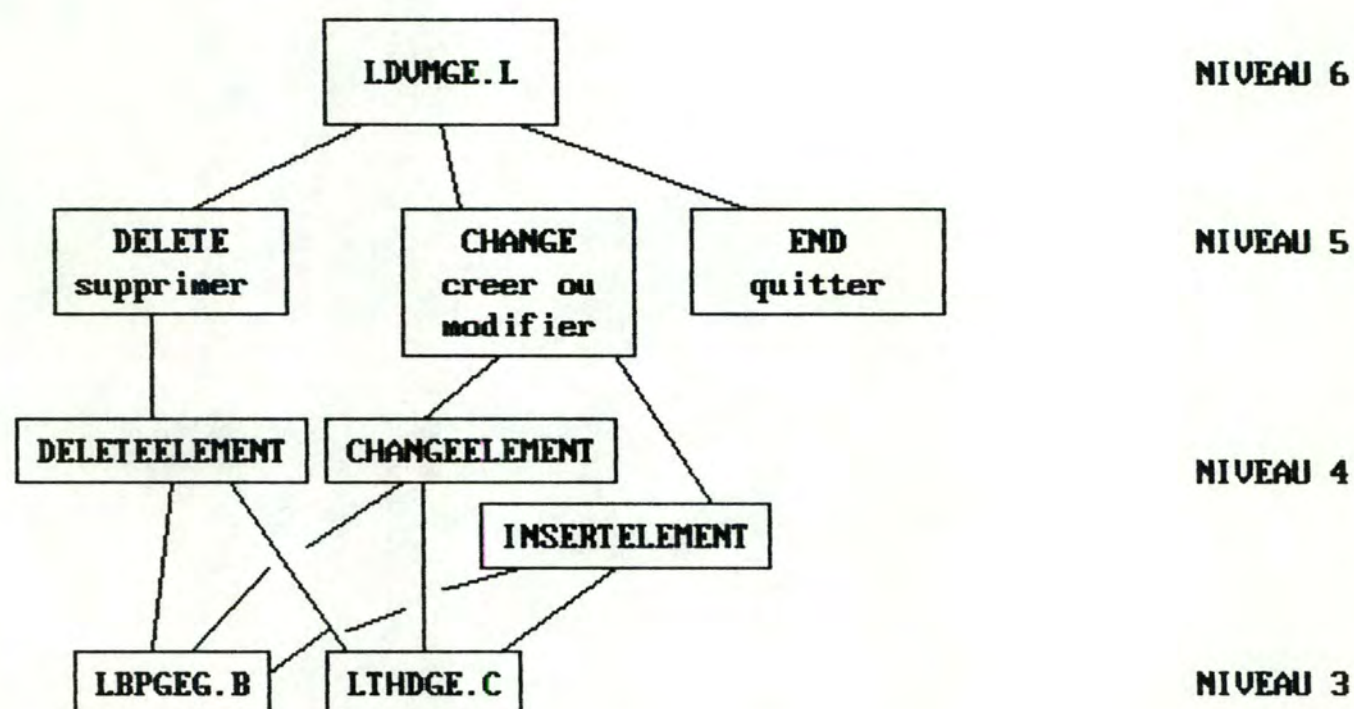
1.2. Exemples de modules

1.2.1. Creation/Maj germes (programmes en Pascal)

1.2.2. Traitement journalier (programmes en Fortran)

2. Schemas des acces necessaires

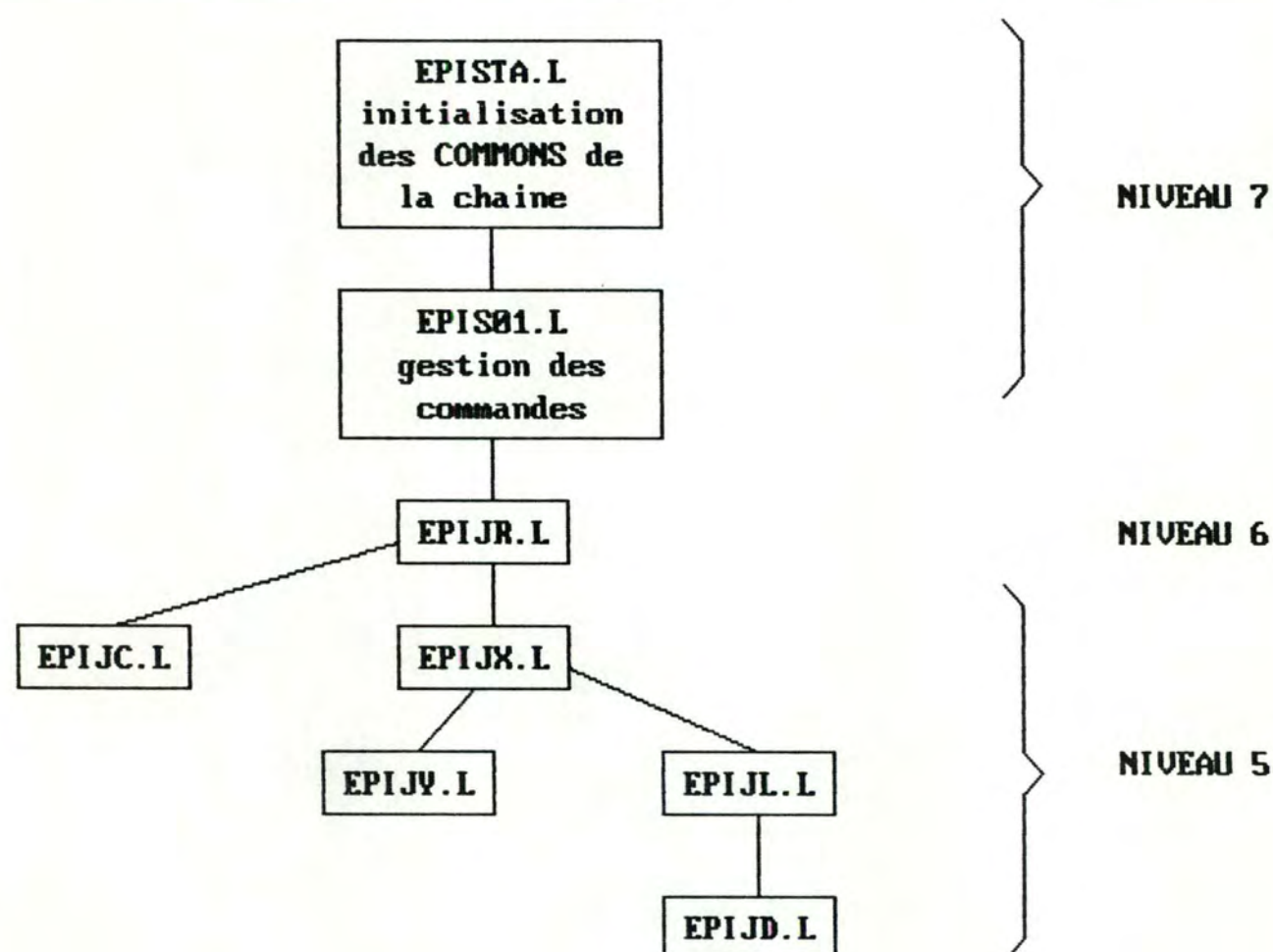
CLIN.St-JEAN	EPI/	REF	UPD.LEU.
LEFEBURE P.	BACGER.DU;PC	DECOUPE -MODULES-GESTION-GERME	



LBPGEGB realise la relation entre les champs de l'ecran et leur description dans le programme

LTHDGE.C decrit le record du THESAURUS des germes

CLIN.St-JEAN	EPI/	REF	UPD.LEV.
LEFEBURE P.	EPI JRN.DU;PC	DECOUPE-MODULES-TRAITEMENT-JRN	



Le role de chacun de ces modules est decrit sous la
forme de commentaires dans les programmes

M O D U L E : L D V M G E . L

```

0001  {[F-]}{
0002          ???
0003
0004  *****
0005
0006  Programma      : LDVMGE.L
0007  Sourcefile     : LDVMGE.D, 332.
0008  T-spec's      :
0009  Auteur         : Pascale Lefebvre
0010  Kreatiedatum   : 04-06-87
0011  Acc./Prod.datum : ...../.....
0012
0013  VERSIE  AUT      DATUM  AARD LAATSTE WIJZIGING  :
0014  -----
0015  Introduction de la version bilingue
0016
0017
0018  *****
0019
0020  FUNKTIEBESCHRIJVING [ LDVMGE.D ]:
0021  -----
0022      Module de gestion du fichier des germes ( TH. 113)
0023
0024
0025
0026  *****
0027          ==BATCH-INFO==
0028
0029
0030
0031  }{[F+]}
0032
0033  {
0034      ;$$
0035      ;$pfgen lbpgeg
0036      ;code_germe
0037      ;genre_germe
0038      ;sp_germe
0039      ;serot_germe
0040      ;forme_germe
0041      ;col_gram_germe
0042      ;lib_fr_germe
0043      ;lib_nl_germe
0044      ;$all i.t,100
0045      ;$pc i.t<LDVMGE
0046      ;$skull LDVMGE.L
0047      ;$oli LDVMGE.L<kb:
0048      ;object: i.t, pasbru.s,pcbouw.s,pcbouw.s, pas.s
0049      ;stop
0050      ;$$
0051  }
0052
0053  {*****}
0054  PROGRAM thesaurusmaintenance;
0055      %INCLUDE 'bosdef.c[*]';
0056      %INCLUDE 'strtst.c[*]';

```



```

0058 %INCLUDE 'brundef.c["*BRU"]';
0059 %INCLUDE 'brundef.e["*BRU"]';
0060 %INCLUDE 'thdefs.c[*]';
0061 %INCLUDE 'lthdgc.c["*LAB"]';
0062 %INCLUDE 'lbpgeg.b["*LAB"]';
0063 %INCLUDE 'pfdefs.c[*]';
0064 %INCLUDE 'cvdefs.c[*]';
0066
0067 CONST
0068     thesnr = 113;
0069     index = 0;
0070     candmax = 128;
0071
0072 TYPE
0073     charset = SET OF char;
0074
0075 VAR
0076     lbpgeg_beeld: lbpgeg_beeldrec;
0077     pf_keyset, pf_inputset: veldset;
0078     candidates: ARRAY [1..candmax] OF long;
0079     elemnr, init_elemnr: long;
0080     oldelement, element: thsger_rec;
0081     searchcrit, init_element: thsger_rec;
0082     masks: maskset;
0083     th_status: th_controllst;
0084     skip_els: short;
0085     action: char;
0086
0087 {=====}
0088     PROCEDURE mparam ;
0089     NONPASCAL;
0090
0091 {-----}
0092     PROCEDURE retprg;
0093     NONPASCAL;
0094
0095 {-----}
0096
0098 { initialize -- initialiseer de sets met de inputvelden en
0099     keyitems, set met maskers en de initiatiewaarden
0100     van de recorditems die niet gemuteerd worden }
0101
0102
0103 PROCEDURE initialize;
0104
0105
0106 BEGIN
0107     init_elemnr := 0;
0108     thread(thesnr, init_element.prm, init_elemnr, th_status);
0109     masks := [];
0110     pf_keyset := [pf_code_germe];
0111     pf_inputset := [pf_genre_germe,
0112                     pf_sp_germe,
0113                     pf_serot_germe,
0114                     pf_forme_germe,
0115                     pf_col_gram_germe,

```

```
0116                pf_lib_fr_germe,  
0117                pf_lib_nl_germe];  
0118    END; { initialize }  
0119  
0120 {-----}  
0121 { tstpfpos -- stel een vraag en test of het antwoord korrekt is,  
0122                zoniet herhaal dan de vraag }  
0123  
0124  
0125 FUNCTION tstpfpos(string: PACKED ARRAY [low..high: short] OF char;  
0126                inpset: charset): char;  
0127  
0128    VAR  
0129        answer: char;  
0130        stridx: short;  
0131  
0132  
0133    BEGIN  
0134        REPEAT  
0135            pfpos;  
0136            FOR stridx := low TO high DO  
0137                write(output, string[stridx]);  
0138            IF eoln(input) THEN  
0139                BEGIN  
0140                    answer := ' ';  
0141                    readln(input)  
0142                END
```


<<SRC>> VOOR USER 332 TYDENS: TESTRU-KSJ L0 11-09-87 19:15 BLAD 4
LDVMGE

```
0143      ELSE
0144      BEGIN
0145      readln(answer);
0146      answer := toupper(answer);
0147      END
```

```
0148     UNTIL answer IN inpset;
0149     pfpos;
0150     tstpfpos := answer;
0151 END; { tstpfpos }
0152
0153 {-----}
0154 { showstatus -- laat status zien }
0155
0156
0157 PROCEDURE showstatus;
0158
0159
0160 BEGIN
0161     pfpos;
0162     CASE th_status OF
0163         th_more:
0164             write(output, 'Meer elementen');
0165         th_thesnr:
0166             write(output, 'Verkeerd thesaurusnummer');
0167         th_length:
0168             write(output, 'Verkeerde lengte');
0169         th_notper:
0170             write(output, 'Aktie niet toegestaan');
0171         th_deleted:
0172             write(output, 'Element is verwijderd');
0173         th_elemnr:
0174             write(output, 'Verkeerd elementnummer');
0175         th_notpos:
0176             write(output, 'Aktie niet mogelijk');
0177         th_full:
0178             write(output, 'Thesaurus is vol');
0179         th_conversion:
0180             write(output, 'Konversie fout');
0181         th_alarm:
0182             write(output, 'Onbekende fout');
0183     END;
0184     write(output, ', typ <CR> voor gezien: ');
0185     readln(input);
0186     pfpos;
0187 END; { showstatus }
0188
0189 {-----}
0190 { screen_to_thesrec -- kopieer het beeldrecord naar het thesaurusrecord }
0191
0192
0193 PROCEDURE screen_to_thesrec;
0194
0195
0196 BEGIN
0197     element.code_th_germe := lbpgeg_beeld.code_germe;
0198     element.genre_th_germe := lbpgeg_beeld.genre_germe;
0199     element.sp_th_germe := lbpgeg_beeld.sp_germe;
0200     element.ser_th_germe := lbpgeg_beeld.serot_germe;
0201     element.forme_th_germe := lbpgeg_beeld.forme_germe;
0202     element.col_th_germe := lbpgeg_beeld.col_gram_germe;
0203     element.lib_fr_th_germe := lbpgeg_beeld.lib_fr_germe;
```



```
0204     element.lib_nl_th_germe := lbpgeg_beeld.lib_nl_germe;
0205     END; { screen_to_thesrec }
0206
0207 {-----}
0208 { thesrec_to_screen -- kopieer het thesaurusrecord naar het beeldrecord }
0209
0210
0211 PROCEDURE thesrec_to_screen;
0212
0213
0214 BEGIN
0215     lbpgeg_beeld.code_germe := element.code_th_germe;
0216     lbpgeg_beeld.genre_germe := element.genre_th_germe;
0217     lbpgeg_beeld.sp_germe := element.sp_th_germe;
0218     lbpgeg_beeld.serot_germe := element.ser_th_germe;
0219     lbpgeg_beeld.forme_germe := element.forme_th_germe;
0220     lbpgeg_beeld.col_gram_germe := element.col_th_germe;
0221     lbpgeg_beeld.lib_fr_germe := element.lib_fr_th_germe;
0222     lbpgeg_beeld.lib_nl_germe := element.lib_nl_th_germe;
0223     END; { thesrec_to_screen }
0224
0225 {-----}
0226 { keyinput -- invoer van de keyitems }
0227
0228
0229 PROCEDURE keyinput(VAR endofinput: boolean);
0230
0231
0232 BEGIN
0233     pfpromark(pf_keyset);
0234     REPEAT
0235         pfask('Key Input');
0236         pferrmark;
0237     UNTIL pfreedy;
0238     endofinput := pfempty(pf_keyset);
0239     IF NOT endofinput THEN
0240         screen_to_thesrec
0241     END; { keyinput }
0242
0243 {-----}
0244 { ischanged -- geeft aan of er minimaal 1 veld
0245             gewijzigd is }
0246
0247
0248 FUNCTION ischanged: boolean;
0249
0250
0251 BEGIN
0252     IF (element.genre_th_germe <> lbpgeg_beeld.genre_germe) OR
0253        (element.sp_th_germe <> lbpgeg_beeld.sp_germe) OR
0254        (element.ser_th_germe <> lbpgeg_beeld.serot_germe) OR
0255        (element.forme_th_germe <> lbpgeg_beeld.forme_germe) OR
0256        (element.col_th_germe <> lbpgeg_beeld.col_gram_germe) OR
0257        (element.lib_fr_th_germe <> lbpgeg_beeld.lib_fr_germe) OR
0258        (element.lib_nl_th_germe <> lbpgeg_beeld.lib_nl_germe) THEN
0259         ischanged := true
```

```
0260     ELSE
0261         ischanged := false
0262     END; { ischanged }
0263
0264 {-----}
0265 { insertelement -- invoer van de elementgegevens en eventueel het
0266                 element toevoegen aan de thesaurus }
0267
0268
0269 PROCEDURE insertelement;
0270
0271
0272 BEGIN
0273     pfpromark(pf_inputset);
0274     REPEAT
0275         pfask('Insert');
0276         pfusrmark;
0277     UNTIL pfready;
0278     IF NOT pfempty(pf_inputset) THEN
0279         BEGIN
0280             element := init_element;
0281             screen_to_thesrec;
0282             thwrite(thesnr, element.prm, elemnr, th_status);
0283             IF th_status <> th_ok THEN
0284                 showstatus;
0285             END;
0286         END; { insertelement }
0287
0288 {-----}
0289 { changeelement -- wijzigen van de elementgegevens en het element
0290                 wijzigen in de thesaurus }
0291
0292
0293 PROCEDURE changeelement;
0294
0295     VAR
0296         answer: char;
0297
0298
0299 BEGIN
0300     oldelement := element;
0301     pfenable(pf_inputset);
0302     REPEAT
0303         pfask('Change');
0304         pfusrmark;
0305     UNTIL pfready;
0306     IF ischanged THEN
0307         BEGIN
0308             screen_to_thesrec;
0309             thupdate(thesnr, element.prm, oldelement.prm, elemnr, th_status);
0310             IF th_status <> th_ok THEN
0311                 showstatus
0312         END
```



```
0313     END; { changeelement }
0314
0315 {-----}
0316 { deleteelement -- laat elementgegevens zien en verwijder evt.
0317               het element uit de thesaurus }
0318
0319
0320 PROCEDURE deleteelement;
0321
0322     VAR
0323         answer: char;
0324
0325
0326     BEGIN
0327         pfenable(pf_inputset);
0328         REPEAT
0329             pfask('Delete');
0330         UNTIL pfready;
0331             user_info_pbru;
0332             CASE user.taal OF
0333                 francais, indefini: answer:=tstpfpos
0334                     ('Etes-vous certain (O/N) ? <CR> = Non: ', ['O', 'N', ' ']);
0335                 neerlandais: answer:=tstpfpos
0336                     ('Weet u het zeker (J/N) ? <CR> = Nee ', ['J', 'N', ' ']);
0337             END;
0338         IF (answer = 'J') OR (answer = 'O') THEN
0339             BEGIN
0340                 thdelete(thesnr, element.prm, elemnr, th_status);
0341                 IF th_status <> th_ok THEN
0342                     showstatus
0343                 END
0344             END
```

```
0345     END; { deleteelement }
0346
0347 {-----}
0348 { change -- wijzigen/invoeren van 1 of meer
0349           elementen }
0350
0351
0352 PROCEDURE change;
0353
0354     VAR
0355         endofinput, stop: boolean;
0356         candidx, nrofels: short;
0357         th_status1: th_controllst;
0358
0359
0360     BEGIN
0361         REPEAT
0362             pfclean(pf_keyset + pf_inputset);
0363             keyinput(endofinput);
0364             IF NOT endofinput THEN
0365                 BEGIN
0366                     searchcrit := element;
0367                     skip_els := 0;
0368                     REPEAT
0369                         thsearch(thesnr, searchcrit.prm, index, masks, candidates,
0370                                 skip_els, th_status);
0371                     IF (th_status <> th_ok) AND (th_status <> th_more) THEN
0372                         BEGIN
0373                             showstatus;
0374                             stop := true;
0375                             END
```


<<SRC>> VOOR USER 332 TYDENS: TESTRU-KSJ L0 11-09-87 19:15 BLAD 10
LDVMGE

```
0376         ELSE
0377             BEGIN
0378                 stop := false;
0379                 nrofels := thcandmod(skip_els, candmax);
0380                 IF nrofels = 0 THEN
0381                     BEGIN
0382                         insertelement;
0383                         stop := true;
0384                     END
```

```
0385         ELSE
0386             BEGIN
0387                 candidx := 0;
0388                 REPEAT
0389                     candidx := candidx + 1;
0390                     elemnr := candidates[candidx];
0391                     thread(thesnr, element.prm, elemnr, th_status1);
0392                     IF th_status1 <> th_ok THEN
0393                         showstatus
0394                     ELSE
0395                         BEGIN
0396                             thesrec_to_screen;
0397                             pfwrite(pf_inputset + pf_keyset);
0398                             changeelement;
0399                         END;
0400                     UNTIL stop OR (candidx = nrofels);
0401                 END;
0402             END;
0403             UNTIL (th_status <> th_more) OR stop;
0404         END;
0405     UNTIL endofinput;
0406 END; { change }
0407
0408 {-----}
0409 { delete -- verwijderen van 1 of meer elementen }
0410
0411
0412 PROCEDURE delete;
0413
0414     VAR
0415         endofinput, stop: boolean;
0416         candidx, nrofels: short;
0417         th_status1: th_controllst;
0418
0419
0420     BEGIN
0421         REPEAT
0422             pfclean(pf_keyset + pf_inputset);
0423             keyinput(endofinput);
0424             IF NOT endofinput THEN
0425                 BEGIN
0426                     searchcrit := element;
0427                     skip_els := 0;
0428                     REPEAT
0429                         thsearch(thesnr, searchcrit.prm, index, masks, candidates,
0430                             skip_els, th_status);
0431                     IF (th_status <> th_ok) AND (th_status <> th_more) THEN
0432                         BEGIN
0433                             showstatus;
0434                             stop := true;
0435                         END
```



```
0436         ELSE
0437             BEGIN
0438                 stop := false;
0439                 nrofels := thcandmod(skip_els, candmax);
0440                 IF nrofels = 0 THEN
0441                     BEGIN
0442                         pfpos;
0443                         user_info_pbru;
0444                         CASE user.taal of
0445                             francais, indefini: write(output,
0446                                 'Aucun candidat retrouve, tapez <CR> ');
0447                             neerlandais : write(output,
0448                                 'Geen kandidaten gevonden, typ <CR> voor gezien: ');
0449                         END;
0450                         readln(input);
0451                         pfpos;
0452                         stop := true;
0453                     END
```

```
0454         ELSE
0455             BEGIN
0456                 candidx := 0;
0457                 REPEAT
0458                     candidx := candidx + 1;
0459                     elemnr := candidates[candidx];
0460                     thread(thesnr, element.prm, elemnr, th_status1);
0461                     IF th_status1 <> th_ok THEN
0462                         showstatus
0463                     ELSE
0464                         BEGIN
0465                             thesrec_to_screen;
0466                             pfwrite(pf_inputset + pf_keyset);
0467                             deleteelement;
0468                         END;
0469                     UNTIL stop OR (candidx = nrofels);
0470                 END;
0471             END;
0472         UNTIL (th_status <> th_more) OR stop;
0473     END;
0474     UNTIL endofinput;
0475     END; { delete }
0476
0477 {=====}
0478 { main -- hoofdprogramma voor het onderhouden van een thesaurus
0479     m.b.v. een pasform beeldje }
0480
0481
0482
0483 BEGIN
0484     mparam;
0485     initialize;
0486     user_info_pbru;
0487     CASE user.taal OF
0488         francais, indefini: pfget('lbpgef', lbpgeg_beeld.prm);
0489         neerlandais: pfget('lbpngen', lbpgeg_beeld.prm);
0490     END;
0491     pfdisplay;
0492     REPEAT
0493         pfenable([]);
0494         REPEAT
0495             pfask('Action')
0496         UNTIL pfready;
0497         user_info_pbru;
0498         CASE user.taal OF
0499             francais, indefini: action:=tstpfpos('Modifier, Supprimer, Quitter :',
0500                 ['M', 'S', 'Q']);
0501             neerlandais: action := tstpfpos('Wijzigen, Verwijderen of Eind: ',
0502                 ['W', 'V', 'E']);
0503         END;
0504         CASE action OF
0505             'W', 'M':
0506                 change;
0507             'V', 'S':
0508                 delete;
0509             'E', 'Q':
0510                 END
```


<<SRC>> VOOR USER 332 TYDENS: TESTRU-KSJ LO 11-09-87 19:15 BLAD 14
LDVMGE

```
0511 UNTIL (action = 'E') OR (action = 'Q');
0512   retprg;
0513 END.
OKAY
```

M O D U L E : L B P G E G . B

<<SRC>> VOOR USER 331 TYDENS: TESTRU-KSJ LO 11-09-87 19:13 BLAD 1
_BPGEG.B

```
0001 TYPE
0002     lbpgeg_beeldrec =
0003     RECORD
0004     CASE short OF
0005     1:
0006     (
0007         code_germe      : {A} PACKED ARRAY [1.. 8] OF char;
0008         genre_germe     : {A} PACKED ARRAY [1..16] OF char;
0009         sp_germe        : {A} PACKED ARRAY [1..16] OF char;
0010         serot_germe     : {A} PACKED ARRAY [1..10] OF char;
0011         forme_germe    : {A} PACKED ARRAY [1..14] OF char;
0012         col_gram_germe : {A} PACKED ARRAY [1..14] OF char;
0013         lib_fr_germe   : {A} PACKED ARRAY [1..32] OF char;
0014         lib_nl_germe   : {A} PACKED ARRAY [1..32] OF char;
0015     );
0016     2: (align : short);
0017     3: (prm : ARRAY [1.. 142] OF char);
0018 END;
0019
0020 pfveld_lst = (
0021     pf_code_germe      ,pf_genre_germe      ,pf_sp_germe      ,pf_serot_germe
0022     ,pf_forme_germe    ,pf_col_gram_germe ,pf_lib_fr_germe ,pf_lib_nl_germe
0023     ,pf_09             ,pf_10             ,pf_11             ,pf_12
0024     ,pf_13             ,pf_14             ,pf_15             ,pf_16
0025     ,pf_17             ,pf_18             ,pf_19             ,pf_20
0026     ,pf_21             ,pf_22             ,pf_23             ,pf_24
0027     ,pf_25             ,pf_26             ,pf_27             ,pf_28
0028     ,pf_29             ,pf_30             ,pf_31             ,pf_32
0029     ,pf_33             ,pf_34             ,pf_35             ,pf_36
0030     ,pf_37             ,pf_38             );
0031 OKAY
```

M O D U L E : L T H D G E . C

<<SRC>> VOOR USER 331 TYDENS: TESTRU-KSJ L0 11-09-87 19:13 BLAD 1
LTHDGE.C

```
0001 TYPE
0002 thsger_rec=
0003
0004 RECORD
0005     CASE SHORT OF
0006     1 :
0007         (code_th_germe : PACKED ARRAY[1..8] OF CHAR;
0008          genre_th_germe : PACKED ARRAY[1..16] OF CHAR;
0009          sp_th_germe : PACKED ARRAY[1..16] OF CHAR;
0010          ser_th_germe : PACKED ARRAY[1..10] OF CHAR;
0011          forme_th_germe : PACKED ARRAY[1..14] OF CHAR;
0012          col_th_germe : PACKED ARRAY[1..14] OF CHAR;
0013          lib_fr_th_germe : PACKED ARRAY [1..32] OF CHAR;
0014          lib_nl_th_germe : PACKED ARRAY [1..32] OF CHAR;
0017          reserve : PACKED ARRAY[1..38] OF CHAR;
0019          );
0020     2 :
0021         (prm : PACKED ARRAY [1..180] OF CHAR);
0022     3 :
0023         (align : short)
0024     END;
0025 OKAY
```

M O D U L E : E P I S T A . L


```
0001 C
0002 C      ???
0003 C
0004 C *****
0005 C
0006 C      PROGRAMMA                : EPISTA.L,23.
0007 C      .D-FILE + USERNR       : EPI500.D, 23.
0008 C      DOCUMENTATIENR          :
0009 C      AUTEUR                   : Pascale Lefebvre
0010 C      GROEP + PROJECT          : LABO : general
0011 C      CREATIEDATUM             : 08-04-1987
0012 C      GEWIJZIGD DOOR          :
0013 C      DATUM LAATSTE WIJZIGING  :
0014 C      AARD LAATSTE WIJZIGING  :
0015 C
0016 C *****
0017 C
0018 C      FUNCTIONELE BESCHRIJVING :
0019 C      MAIN-module chaine EPISTA
0020 C      (initialisations des commons et mise en page)
0021 C      GEBRUIKSAANWIJZING      :
0022 C
0023 C *****
0024 C
0025 C$$$
0026 C$PAG 0
0027 C$FC EPI500/K/5
0028 C$ALL P.T,100
0029 C$LINK EPISTA
0030 CP.T,KB:/SH<I.T/CC,ALO,AL1,BRA,BAS/E
0031 C$MANIPO EPISTA.L/K
0032 C
0033 C$LOPFIL EPISTA<P.T
0034 C$PAG 24
0035 C$$$
0036 C
0037 C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
0038 C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
0039 C
0040 C      BLOCK DATA OVRLAY
0041 C      COMMON /RECOVL/ Common systeme I/O et overlay
0042 C      COMMON /MENU / Informations generales JOB/BRU
0043 C      COMMON /EBFPRM/ Parametres epidemio.
0044 C
0045 C      COMMON /EBFDTA/ Informations datas edition
0046 C      END
```

```
0047 C  
0048 C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  
0049 C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  
0050 C  
0051 C  
0052 PROGRAM EPISTA  
0053 C  
0054 COMMON /RECOVL/IRCOVL(65)  
0055 COMMON /MENU /ISTR(80)  
0056 COMMON /EBFPRM/IBFPRM(32)  
0057 C  
0058 COMMON /EBFDA/BBFDA(40)  
0059 C  
0060 C  
0061 REAL*8 RDHEAD(5,2)  
0062 DIMENSION IBUF(256),JBUF(8)  
0063 BYTE BBFDA,BHLP(72),BBUF(144)  
0064 C  
0065 EQUIVALENCE (ISTR(38),IDEV),(ISTR(46),ILA),(ISTR(55),ILAN)  
0066 *,(ISTR(51),IORG),(ISTR(71),ITERM)  
0067 EQUIVALENCE (IBFPRM(1),IBPRED)  
0068 EQUIVALENCE (IBUF(1),BBUF(1))  
0069 C  
0070 DATA RDHEAD/'EPIDEMIO','LOGIE : ','LISTES E','T STATIS','TIQUES '  
0071 *,'EPIDEMIO','LOGIE : ','LIJSTEN ','EN STATI','STIEKEN '/  
0072 DATA IDDRED /7/  
0073 C  
0074 C -----  
0075 C  
0076 CALL MPARAM(ITPRUN,ISTR,IEXCEP,IBUF) ! Gestion et recept. "GO"  
0077 C  
0078 C *****  
0079 C *****  
0080 C  
0081 CALL FIOBUF(BHLP,LENG) ! Recept. info. param. prog.  
0082 C  
0083 C *****  
0084 C  
0085 C+++ Remplissage du COMMON /MENU/  
0086 C  
0087 JBUF(8)=0 ! Init. subr. USMENU  
0088 CALL USMENU(IBUF,JBUF) ! Remplissage common /MENU /  
0089 CALL JOBSIM(BHLP,LENG,BBUF,0) ! Gestion info. param. prog.  
0090 C  
0091 C++ Lire PARAM epidemio.(numero 1)  
0092 C  
0093 CALL RPRMR2('LAB ',890,KONTRL,IBFPRM,64,1)  
0094 IF(KONTRL.EQ.0)GOTO 10 ! Test si lecture Ok  
0095 CALL DBERR(1,1,18,0,0,1,KONTRL,BBUF,N) ! Err. qcq de lecture  
0096 CALL ERRIN('.MAIN.',BBUF,4,1) ! Write fichier des messages  
0097 IBPRED=IDDRED ! Delai de redondance default  
0098 C  
0099 C++ Initialisation des variables et de la chaine  
0100 C++ Mise en page  
0101 C  
0102 10 CALL RPARDA(IORG,16+ILAN,IBUF,KONTRL) ! Lecture data titre edit.
```


<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 3
EPI500

```
0103      CALL MOVCP( BBFDTA, IBUF, 40)          ! Mov ds common /EBFDTA/
0104 C
0105 C *****
0106 C
0107      CALL SPATOF( 40, RDHEAD( 1, ILAN), KONTRL) ! Long. eff. nom programme
0108      CALL PHOOFD( RDHEAD( 1, ILAN), KONTRL, 1, BBUF) ! edit. nom du programme
0109      CALL SAY( ' ')          ! Une ligne de space
0110 C
0111      IDEV=5          ! Init. num. log. si spool
0112      LENG=MOVCP( BHLP, 'D', 1)          ! Init. subr. TERM (term. def.)
0113      CALL MOVCP( JBUF, ' ', -12)          ! " " " edit.
0114      KONTRL=1          ! " " " output
0115      CALL TRM( 'TERM', BHLP, LENG, ILA, 'P1', JBUF, IBUF, IDEV, ITERM, KONTRL)
0116      IF( KONTRL.LE.0) ITERM=0          ! test si OK
0117 C
0118 C *****
0119 C
0120      CALL CLORUN( '/EPI51.L' )          ! Run vers JOB (??)
0121 C
0122 C
0123      END
OKAY
```

M O D U L E : E P I S O 1 . L


```
0001 C  
0002 C      ???  
0003 C  
0004 C *****  
0005 C  
0006 C      PROGRAMMA                : EPI501.L  
0007 C      .D-FILE + USERNR        : EPI501.D, 23.  
0008 C      DOCUMENTATIENR           :  
0009 C      AUTEUR                    : Pascale Lefebvre  
0010 C      GROEP + PROJECT            : Epidemio.  
0011 C      CREATIEDATUM              : 07-04-1987  
0012 C      GEWIJZIGD DOOR           :  
0013 C      DATUM LAATSTE WIJZIGING   :  
0014 C      AARD LAATSTE WIJZIGING    :  
0015 C  
0016 C *****  
0017 C  
0018 C      FUNCTIONELE BESCHRIJVING :  
0019 C      Gestion des differentes commandes de l'epidemio.  
0020 C      - Commandes generales : TERM, KB, SOS, HELP, etc ...  
0021 C      - Commande specifique a l'epidemio. : LIST  
0022 C  
0023 C      GEBRUIKSAANWIJZING        :  
0024 C      - Commande list  
0025 C  
0026 C *****  
0027 C  
0028 C      GEBRUIKTE BOSFORMS        :  
0029 C  
0030 C *****  
0031 C  
0032 C$$  
0033 C$PAG 0  
0034 C$FC EPI501/K/S  
0035 C$ALL P.T,100/ZE  
0036 C$LINK /EPI51  
0037 CP.T,KB:/SH<I.T/CC,LABO,ALO,BRA,BAS/E  
0038 C$MANIPO /EPI51.L/K  
0039 C  
0040 C$LOPFOV /EPI51.L<P.T  
0041 CJ  
0042 C$PAG 24  
0043 C$$  
0044 C  
0045 C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  
0046 C$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$  
0047 C  
0048 BLOCK DATA OVRLAY  
0049 COMMON /RECOVL/  
0050 COMMON /MENU /  
0051 COMMON /EBFPRM/  
0052 COMMON /EBFDTA/  
0053 END
```

[illegible]

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 3
EPI501

```
0110 C *****
0111 C
0112 C++      KB / TERM
0113 C
0114      30 KONTRL=1                      ! Init. subr. TERM (echo)
0115      CALL MOVCP(LBBUF,32,-12)         ! " " " "
0116      CALL TRM(COMMAN(IAD),BHLP,LENG,ILA,'D1',BBUF,IBUF,IDEV,ITERM,KONTRL
0117      IF(KONTRL.LE.0)GOTO 10            ! Test si err. qcq
0118      IF(IBUF(3).GT.2)IDEV=5            ! Si term. type spl = Ok
0119      GOTO 10
0120 C
0121 C *****
0122 C
0123 C+      FRAN / NEDE
0124 C
0125      40 ILA=IAD - 5                    ! Langue : var. 0/1 = Fr/NL
0126      ILAN=ILA + 1                      ! " : var. 1/2 = Fr/NL
0127      CALL RWRISP(6,RDDIAL(1,ILAN),24) ! Write KB type de langue
0128      GOTO 10
0129 C
0130 C *****
0131 C
0132 C++      LIST
0133 C
0134      140 IF(LENG.LE.0)GOTO 150          ! Test si infos. avec commande
0135      IAUTR=1                            ! Init. subr. GET
0136      CALL GETQ(BHLP,LENG,ITYP,INT,DI,R,BS,0,771) ! Read KB
0137      IF(LENG.EQ.1.AND.ITYP.EQ.1)GOTO 170
0138 C
0139 C++      Echo des possibilites (commande 'LIST')
0140 C
0141      150 DO 160 IID=1,IMXPRG
0142      CALL RWRISP(6,BLINE,IEDIT(BLINE)+IST(91)+IIA(2,IID)+IST(93)
0143      * +IAA(165,RDQUES(1,IID,ILAN)))
0144      160 CONTINUE
0145 C
0146 C++      Numero ?
0147 C
0148      CALL RWRISP(6,RDNUMR(1,ILAN),16,2)
0149      IAUTR=1
0150      CALL GET(ITYP,INT,DI,R)
0151      GOTO(170,150,150,150,10),ITYP
0152 C
0153 C++      Decision de run selon le numero selectionne
0154 C
0155      170 IF(INT.LT.1.OR.INT.GT.IMXPRG)GOTO 150
0156      CALL PLEXIF(INT,KONTRL,1,2,3)
0157      GOTO(180,185,190),KONTRL
0158 C
0159      180 CALL CLORUN ('/EPIJR.L')        ! Listes journalieres
0160      GOTO 10
0161      185 CALL CLORUN ('/EPIIX.L')        ! Listes diverses
0162      GOTO 10
0163      190 CALL CLORUN ('/EPISS.L')       ! Listes statistique
0164      GOTO 10
0165 C
```

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ KO 11-09-87 16:50 BLAD 4
EPIS01

```
0166 C *****
0167 C
0168 C++ STOP
0169 C
0170 220 CALL SAY ('???')
0171 CALL EXIT
0172 C
0173 C++ BYE
0174 C
0175 230 CALL BYE
0176 C
0177 C
0178 END
OKAY
```


M O D U L E : E P I J R . L

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 1
EPIOJR

```
0001 C
0002 C      ???
0003 C
0004 C *****
0005 C
0006 C      PROGRAMMA                      : /EPIJR.L
0007 C      .D-FILE + USERNR              : EPIOJR.D, 23.
0008 C      DOCUMENTATIENR                 :
0009 C      AUTEUR                          : Pascale Lefebvre
0010 C      GROEP + PROJECT                  : Epidemiologie, selection rec.ds JNR
0011 C      CREATIEDATUM                     : 02-06-1987
0012 C      GEWIJZIGD DOOR                  :
0013 C      DATUM LAATSTE WIJZIGING         :
0014 C      AARD LAATSTE WIJZIGING          :
0015 C
0016 C *****
0017 C
0018 C      FUNCTIONELE BESCHRIJVING :
0019 C
0020 C      Module qui selectionne dans le journalier les records a tester
0021 C      pour la pathogenicite ou cibles.
0022 C      Records selectionnes : record avec bit 8 sans bit 1 et 2.
0023 C
0024 C      GEBRUIKSAANWIJZING             :
0025 C
0026 C *****
0027 C
0028 C
0029 C $$
0030 C $PAG 0
0031 C $FC EPIOJR/K/S
0032 C $ALL P.T,100/ZE
0033 C $LINK /EPIJR
0034 C P,KB:/SH<I.T/CC,LABO,EPI
0035 C ,LCO,ALO,AL1,BRA,BAS/E
0036 C $MANIPO /EPIJR.L/K
0037 C
0038 C $LOPFOV /EPIJR.L<P.T
0039 C JA
0040 C $PAG 24
0041 C $$
0042 C
0043 C *****
0044 C
0045 C      BLOCK DATA OVLAY
0046 C      COMMON /RECOVL/      ! Common systeme I/O et overlay
0047 C      COMMON /MENU /       ! Informations generales JOB/BRU
0048 C      COMMON /EBFPRM/      ! Informations parametres
0049 C      COMMON /EBFDA/       ! Informations datas edition
0050 C
0051 C      COMMON /EBFTPJ/      ! Definition .T records select. "journalier"
0052 C      END
```


[illegible]


```

0109      CALL ROPER(ILUTPJ,RNMTPJ,IMXTPJ,ILNTPJ,IVATPJ)      ! Ouverture .
0110      C
0111      C++          Lecture du journalier
0112      C++          Test si record accepte
0113      C
0114      20 CALL RREATB(8,ITHJNR,28,DTJNRR,KONTRL) ! Lecture thes. journalier
0115      IF(KONTRL.EQ.-1)GOTO 70      ! Test si fin de fichier
0116      C
0117      IF(ITJBTG.EQ.-1)GOTO 20      ! Test si doublet
0118      IF(.NOT.LTEST(7,ITJBTG))GOTO 20      ! Test si deja "mov" ds epi.
0119      IF(LTEST(0,ITJBTG))GOTO 30      ! Test si bit 1 mis (path.)
0120      IF(LTEST(1,ITJBTG))GOTO 30      ! Test si bit 2 mis (cible)
0121      C
0122      C+      Test si resultat positif
0123      C+      Test si patient ambulant
0124      C
0125      IF(ITTEST(DTJGER).EQ.0)GOTO 30      ! Test si germe <> 0
0126      IF(ITTEST(DTJNAD).NE.0)GOTO 60      ! Test si num.admission <> 0
0127      C
0128      C *****
0129      C
0130      C++      Candidats refuses
0131      C+      Update thes. journalier
0132      C
0133      30 N=0      ! Init. compt. multi-users
0134      40 CALL MOVCPL(IBUF,ITHJNR,56)      ! Mov OLD/update
0135      CALL SET(0,ITJBTG)      ! Set flag "pathogene" Ok
0136      CALL SET(1,ITJBTG)      ! " " "cible" Ok
0137      IF(IEQUAL(ITHJNR,IBUF,56).EQ.0)GOTO 20 ! Test si updt. necessaire
0138      CALL DUPTHE(NTHJNR,KONTRL,ITHJNR,IBUF,DTJNRR) ! Update
0139      IF(KONTRL.EQ.0)GOTO 20      ! test si update Ok
0140      IF(KONTRL.NE.-3.OR.N.GT.3)GOTO 50      ! Test si err. multi-users
0141      CALL LRDJNR(DTJNRR,56,1,KONTRL)      ! Err. multi-users (impos. !!!)
0142      N=N + 1      ! Increment. compt. multi-user
0143      GOTO 40
0144      50 CALL DBERR(INTT(DTJNRR),1,NTHJNR,0,0,3,KONTRL,BBUF,N)
0145      CALL ERRIN('MAIN.',BBUF,4,1)      ! Erreur qcq update
0146      GOTO 20
0147      C
0148      C *****
0149      C
0150      C++      Record journalier accepte
0151      C      write .T "J"
0152      C
0153      60 CALL LRDBYS(DTJBYS,140,1,KONTRL)      ! Lecture "BS" (body-site)
0154      IF(KONTRL.NE.0)GOTO 20      ! Test si lecture Ok
0155      OPTMST=DTYNRR      ! Point. "BS"
0156      IF(ITTEST(DTYMST).NE.0)OPTMST=DTYMST ! Si "slave", point.="master"
0157      CALL MOVCPL(IBFTPJ,OPTMST,4,DTJGER,4,0,-48,DTJNRR,4)
0158      CALL RWRIRS(ILUTPJ,IBFTPJ,ILNTPJ,IVATPJ) ! Write .T
0159      GOTO 20
0160      70 IF(IVATPJ.EQ.1)GOTO 90      ! Test si au moins 1 record
0161      CALL MOVCPL(IBFTPJ,'0',-ILNTPJ)      ! EOF
0162      CALL RWRIRS(ILUTPJ,IBFTPJ,ILNTPJ,IVATPJ) ! Write EOF
0163      C
0164      C++      Tri du fichier temporaire (tri s/ Body-site et germe)

```


<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 4
EPIOJR

```
0165 C
0166 CALL RSORT(ILUTPJ,RNMPJ,'1,U,1,A,5,U,1,A/ ',IRP)
0167 IF(IRP(1).EQ.0)GOTO 80 ! Test si tri Ok
0168 CALL DBERR(0,0,0,0,0,6,IRP,BBUF,N) ! Erreur qcq tri
0169 CALL ERRIN('MAIN.',BBUF,4,1)
0170 GOTO 100
0171 80 IMXTPJ=IRP(2) ! Nombre effectif de records
0172 IVATPJ=1 ! Reinit. variable associee
0173 C
0174 C++ Appel module EPIOJX qui traite pathogenicite et/ou cible
0175 C
0176 CALL CLORUN('/EPIJX.L') ! Run mod. test si P/C
0177 C
0178 C *****
0179 C *****
0180 C
0181 90 CALL RWRISP(6,RDNEOT(1,ILAN),72) ! Aucun record path./cible
0182 100 CALL CLORUN('/EPIS1.L') ! Retour a JOB (??)
0183 C
0184 C
0185 END
OKAY
```

M O D U L E : EPIJC.L


```
C010 C
C020 C      ???
C030 C
C040 C *****
C050 C
C060 C      PROGRAMMA                      : /EPIJC.L
C070 C      .D-FILE + USERNR              : EPIOJC.D, 23.
C080 C      DOCUMENTATIENR                :
C090 C      AUTEUR                        : Pascale Lefebvre
C100 C      GROEP + PROJECT                : Epidemiologie, cont. fich. journalier
C110 C      CREATIEDATUM                  : 19-05-1987
C120 C      GEWIJZIGD DOOR                :
C130 C      DATUM LAATSTE WIJZIGING       :
C140 C      AARD LAATSTE WIJZIGING        :
C150 C
C160 C *****
C170 C
C180 C      FUNCTIONELE BESCHRIJVING :
C190 C      Premier module des listes journalieres des germes
C200 C      dont la fonction est de tester les records.
C210 C      1- Controle si certains records du journalier sont deja "moves".
C220 C      Si non, write dans fichier temporaire (numero de demande labo
C230 C      date et heure + le numero de record)
C240 C      3- Tri sur info. demande labo.
C250 C      4- Relecture .T et test si presence de records "doublets"
C260 C      si oui, update systematique du thesaurus "journalier"
C270 C      si non, write du record en TVR derriere le th.epidemio
C280 C      update record thesaurus EPI (date dernier write)
C290 C      6- Fermeture des fichiers et release fichier temporaire
C300 C
C310 C      GEBRUIKSAANWIJZING              :
C320 C      Programme EPISTA, liste 1.
C330 C
C340 C *****
C350 C
C360 C
C370 C $$
C380 C $PAG 0
C390 C $FC EPIOJC/K/S
C400 C $ALL P.T,100/ZE
C410 C $LINK /EPIJC
C420 C P,KB:/SH<I.T/CC,LABO,EPI
C430 C ,LCO,ALO,AL1,BRA,BAS/E
C440 C $MANIPO /EPIJC.L/K
C450 C
C460 C $LOPFOV /EPIJC.L<P.T
C470 C JA
C480 C $PAG 24
C490 C $$
C500 C
C510 C *****
C520 C
C530 C      BLOCK DATA OVRLAY
C540 C
C550 C      COMMON /RECOVL/      ! Common systeme I/O et overlay
```

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 2
EPIOJC

C560 COMMON /MENU / ! Informations generales JOB/BRU
C570 COMMON /EBFPRM/ ! parametres epidemio.
C580 C
C590 END

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 4
EPIOJC

```

160 C      Numero demande (D/T)      1      1      8
170 C      Date de la demande      5      9      2
180 C      Heure " " "      6      11      2
190 C      Num. rec. JNR      7      13      4
200 C
210 C -----
220 C
230 C      CALL MPARAM                      ! Module appele en SUBPRG
240 C
250 C *****
260 C *****
270 C
280 C      CALL RWRISP(6,RDEDIT(1,ILAN),72)      ! Write message KB
290 C
300 C++      Initialisations
310 C
320 C      RNMT PJ='A.T '                      ! Nom du fich. temp. work
330 C      ILUT PJ=9                          ! Numero logique .T
340 C      IMXTPJ=0                          ! Nombre max. records accepte
350 C      ILNTPJ=16                          ! Long. (bytes) records .T
360 C      IVATPJ=1                          ! Init. var. associee
370 C      CALL SETTMR(RNMT PJ,200)          ! Def. .T (200 secteurs)
380 C
390 C      CALL RALBUF(8,-1,ILUT PJ,4)          ! Attribution des buffers
400 C      CALL ROPETH(8,NTHJNR)              ! Relation num. log./th. jour
410 C      CALL ROPER(ILUT PJ,RNMT PJ,IMXTPJ,ILNTPJ,IVATPJ) ! Relation num./ .T
420 C
430 C++      Lecture du journalier
440 C++      Ecriture dans fichier temporaire des records non encore "moves"
450 C
460 C      10 CALL RREATB(8,ITHJNR,28,DTJNRR,KONTRL) ! Lecture thes. journalier
470 C      IF(KONTRL.EQ.-1)GOTO 30              ! Test si fin de fichier
480 C      IF(LTEST(7,ITJBTG))GOTO 10          ! Test si deja "move" ds epi.
490 C
500 C      IF(IVATPJ.GT.IMXTPJ)GOTO 20          ! Test si depasse max. .T
510 C      CALL MOVCPJ(IBFTPJ,DTJNRD,8,ITJDTW,2,ITJHRW,2,DTJNRR,4)
520 C      CALL RWRIRS(ILUT PJ,IBFTPJ,ILNTPJ,IVATPJ) ! Write .T
530 C      GOTO 10
540 C
550 C      20 N=IEDIT(BBUF)+ITB(49)+IST('Max. records work file')+IIA(8,IMXTPJ)
560 C      *+ITB(141)
570 C      CALL ERRIN(' .MAIN. ',BBUF,4,1)
580 C      30 IF(IVATPJ.EQ.1)GOTO 360          ! Test aucun record trouve
590 C      CALL MOVCPJ(IBFTPJ,'0',-ILNTPJ)      ! Record EOF .T
600 C      CALL RWRIRS(ILUT PJ,IBFTPJ,ILNTPJ,IVATPJ) ! Write EOF
610 C      CALL RFORCE(ILUT PJ)                ! Write force .T disque
620 C      CALL RALBUF                          ! Fermeture des buffers
630 C
640 C++      Tri du .T (Numero tdemande (T/D), date et heure demande)
650 C
660 C      CALL DSORT(RNMT PJ,RNMT PJ,ILNTPJ
670 C      *, '5,U,1,A,1,U,1,D,9,I,1,D,11,I,1,D/' ,IRP)
680 C      IF(IRP(1).EQ.0)GOTO 40              ! Tri et test Ok
690 C      CALL DBERR(0,0,0,0,0,6,IRP,BBUF,N)  ! Erreur qcq tri
700 C      CALL ERRIN(' .MAIN. ',BBUF,4,1)      ! Write fichier erreurs
710 C      GOTO 370                            ! Le prog. est stoppe !!!
720 C      40 IMXTPJ=IRP(2)                    ! Nombre de records .T

```



```

1730          IVATPJ=1                                ! Reinit. var. associee
1740 C
1750 C *****
1760 C
1770 C++          Lecture .T et write records epidemio et /ou tvr epidemio.
1780 C
1810          CALL RALBUF(ILUTPJ,-1)                    ! Redef. buffer (lecture)
1820          CALL ROPER(ILUTPJ,RNMTPJ,IMXTPJ,ILNTPJ,IVATPJ) ! Relation num./ .T
1830 C
1840          CALL MOVCP(LIBFCMP,0,-12)                 ! Init. zone de comp. OLD
1850          50 IF(IVATPJ.GT.IMXTPJ)GOTO 370             ! test si fin .T
1860          CALL RREAR(ILUTPJ,IBFTPJ,ILNTPJ,IVATPJ) ! Lecture .T
1870          CALL LRDJNR(DBFJNR,56,1,KONTRL)           ! Read thes. JNR
1880          IF(KONTRL.NE.0)GOTO 50                     ! Test si lecture Ok
1890 C
1900 C+          Test si rupture de demande
1910 C
1920          LDBL=.FALSE.                               ! Logical doublet=.true.
1930          IF(IEQUAL(IBFTPJ,DNDCMP,8).NE.0)GOTO 60 ! Test si demande diff.
1940          IF(IBFTPJ(5).NE.IDTCMP)GOTO 60             ! Test si date differente
1950          IF(IBFTPJ(6).LT.IHRCMP)GOTO 300            ! Test si heure inferieur
1960 C
1970 C+          Search record epidemio.
1980 C+          Si pas trouve, write record epidemio
1990 C+          Read TVR epidemio de controle (delete si trouve)
2000 C+          Write TVR epidemio
2010 C
2020          60 LUPDT=.FALSE.                           ! Record epi. retrouve
2030          IDLRED=0                                    ! Init. delai redondance
2040          IFLNRD=0                                    ! " flag non redondant
2050 C
2060          CALL LRDBYS(DTJBYS,140,1,KONTRL)           ! Lecture "BS"
2070          IF(KONTRL.NE.0)GOTO 50                     ! Test si lecture Ok
2080          DPTMST=DTYNRR                               ! Numero record "BS"
2090          IF(ITTEST(DTYMST).NE.0)DPTMST=DTYMST ! Si "slave", attr. "master"
2100          DTENRR=DSREPI(DPTMST,DTJGER,DTJPAT,BBUF) ! Search rec.epi
2110          IF(ITTEST(DTENRR))50,70,80                 ! <0:error,=0:not trouve,>0:OK
2120 C
2130          70 IF(ITTEST(DTJGER))90,180,90             ! Test si culture neg.
2150          80 CALL LRDEPI(DTENRR,32,1,KONTRL)         ! Lecture thes. epidemio.
2160          IF(KONTRL.NE.0)GOTO 50                     ! test si lecture Ok
2170          LUPDT=.TRUE.                               ! Mode update
2180          IF(ITTEST(DTJGER).EQ.0)GOTO 200            ! Test si culture neg.
2190 C
2200 C++          Lecture du delai de non redondance pour ce germe dans ce body-site
2210 C          1. Voir dans TVR particularite germe ds B-S (TVR 115.1) germe > 0
2220 C          2. Voir dans TVR generalite body-site (TVR 115.1) germe = 0
2221 C          3. Valeur de default = parametre epi.
2230 C
2240          90 CALL MOVCP(LDTTBDG,DTJGER,4,0,-4)       ! Init. zone point. germes
2250          DO 130 IDO=1,2                             ! Cycle germe "germe" ou "*"
2260          KONTRL=1                                    ! Init. DRPVR3
2270          NSKIP=0                                     ! " "
2290          CALL DRPVR3(DPTMST,KONTRL,IBFSBR,2048,NSKIP,1,DTTBDG(IDO),4,0
2300          * ,NTHBYS,0,DPTMST)
2310          IF(KONTRL)100,130,110                       ! Test si retrouve

```


<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 6
EPIOJC

```
2320 100 IF(KONTRL.EQ.-4)GOTO 130 ! Test si jamais ecrit
2330 CALL DBERR(INTT(DPTMST),2,NTHBYS,1,0,1,KONTRL,BBUF,N)
2340 CALL ERRIN('MAIN.',BBUF,4,1) ! Erreur qcq lecture
2350 GOTO 130
2360 110 IF(KONTRL.EQ.1)GOTO 120 ! Max. un seul subsubrecord
2370 N=IEDIT(BBUF)+ITB(49)+IST('MULTIPLE SUBSUBRECORD "BODY-SITE"')
2380 * +ITB(141)
2390 CALL ERRIN('MAIN.',BBUF,4,1) ! >1 subsubrecord, continue
2400 GOTO 130
2410 120 IDLRED=IBF5BR(6) ! Delai de redondance "G"/"*"
2420 GOTO 140
2430 130 CONTINUE
2440 IDLRED=IBPREO ! Delai de redondance "P"
2450 C
2460 C *****
2470 C
2480 140 IF(.NOT.LUPDT)GOTO 170 ! Test si mode "update"
2490 C
2500 C+ Test de redondance
2510 C
2520 IF(ITJDAT-ITEDTM.LE.0)GOTO 200 ! date adm. - date "mere"
2530 IF(ITJDAT-ITEDTM.GE.IDLRED)GOTO 190 ! Test si redondant
2540 DI=DSREPI(DPTMST,IDTNUL,DTJPAT,BBUF) ! Recherche si negatif inter.
2550 IF(ITTEST(DI))50,200,160 ! <0=err,0=non trouve,>0 Ok
2560 160 CALL MOVCP(LIBUF(101),ITHEPI,36) ! Sav record epidemio.
2570 CALL LRDEPI(DI,32,1,KONTRL) ! Lecture rec. epi. NIEW "mere"
2580 CALL MOVCP(LIBUF,ITHEPI,36) ! Remov record premier
2590 CALL MOVCP(LITHEPI,IBUF(101),36) ! Mov ds OLD/update
2600 IF(IBUF(9).LT.ITJDAT.AND.IBUF(9).GT.ITEDTM)GOTO 190
2630 GOTO 200
2680 C
2690 C *****
2700 C
2710 C+ Write record epidemio.
2720 C
2730 170 IFLNRD=1 ! Flag non redondant "true"
2740 180 CALL MOVCP(LITHEPI,DPTMST,4,DTJGER,4,DTJPAT,4,ITJBTG,2,ITJDAT,2
2750 *,ITJDAT,2,ITJDAT,2,DTJNRD,8,IDLREN,2,0,-2)
2760 CALL DWRTHE(NTHEPI,KONTRL,ITHEPI,DTENRR) ! Write record EPI
2770 IF(KONTRL.EQ.0)GOTO 290 ! Controle write
2780 CALL DBERR(INTT(DTENRR),2,NTHEPI,1,0,2,KONTRL,BBUF,N)
2790 CALL ERRIN('MAIN.',BBUF,4,1) ! Erreur qcq write
2800 GOTO 350
2810 C
2820 C *****
2830 C
2840 C+ Update record epidemio (date de dernier write)
2850 C
2860 190 IFLNRD=1 ! Flag non redondant "true"
2870 200 N=0 ! Init. compteur multi-users
2880 210 CALL MOVCP(LIBUF,ITHEPI,36) ! Mov rec. epi. OLD/update
2890 IF(ITJDAT.LT.ITEDTL)GOTO 220 ! Test "JNR" < dernier "wagon"
2900 ITEDTL=ITJDAT ! Attrib. date dernier write
2910 GOTO 230 !
2920 220 IF(ITJDAT.GE.ITEDTF)GOTO 230 ! Test "JNR" > 1ier "wagon"
2930 ITEDTF=ITJDAT ! Attrib. date premier write
```


<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 7
EPIOJC

```
2940 230 IF(IFLNRD.EQ.0)GOTO 240 ! Test si "mere" (non redend.)
2950 ITEDLR=IDLRED ! oui, attrib. delai en cours
2960 ITEDTM=ITJDAT ! " " date dniere mere
2970 DTEDMR(1)=DTJNRD(1) ! " " dem. " "
2980 DTEDMR(2)=DTJNRD(2) ! " " " " "
2990 240 IF(IEQUAL(ITHEPI,IBUF,36).EQ.0)GOTO 260 ! Test si updt. necessaire
3000 CALL DUPTHE(NTHEPI,KONTRL,ITHEPI,IBUF,DTENRR) ! Update
3010 IF(KONTRL.EQ.0)GOTO 260 ! Test si update Ok
3020 IF(KONTRL.NE.-3.OR.N.GT.3)GOTO 250 ! Test si update multi-users
3030 N=N + 1 ! Increment. compt. multi-users
3040 GOTO 210
3050 250 CALL DBERR(INTT(DTENRR),1,NTHEPI,0,0,3,KONTRL,BBUF,N)
3060 CALL ERRIN('MAIN.',BBUF,4,1) ! Erreur qcq update
3070 GOTO 50
3080 C
3090 C *****
3091 C
3092 C++ Lecture, si ce subsubrecord existe deja, suppression de celui-ci
3093 C et ecriture du nouveau
3094 C
3100 260 KONTRL=1 ! Egalite des FFP's
3110 NSKIP=0 ! Initialisation DRPVR3
3120 CALL DRPVR3(DTENRR,KONTRL,IBFSBR,1024,NSKIP,1,DTJNRD,8,6,NTHEPI,0
3130 *,DTENRR)
3140 IF(KONTRL)270,290,280 ! <0:err.,=0:not OK,>0:OK
3150 270 IF(KONTRL.EQ.-4)GOTO 290 ! Test si jamais ecrit
3160 CALL DBERR(INTT(DTENRR),2,NTHEPI,1,0,1,KONTRL,BBUF,N)
3170 CALL ERRIN('MAIN.',BBUF,4,1) ! Erreur qcq lecture
3180 GOTO 50
3190 C
3200 C+ Si TVR trouve, delete TVR
3210 C
3220 280 CALL DEPVR(DTENRR,KONTRL,IBFEPI,NTHEPI,0,DTENRR) ! Delete TVR
3230 IF(KONTRL.EQ.0)GOTO 290 ! Test si delete OK
3240 CALL DBERR(INTT(DTENRR),2,NTHEPI,TYP,0,4,KONTRL,BBUF,N)
3250 CALL ERRIN('MAIN.',BBUF,4,1) ! Erreur qcq delete
3260 GOTO 50
3270 C
3280 C+ Write TVR epidemio
3290 C
3300 290 CALL MOVCP(1BFEPI,ITJDTA,2,ITJDAT,2,DTJNAD,4,DTJNRD,8,ITJCTD,2
3310 *,ITJSRV,2,ITJCHB,2,DTJBYS,4,ITJGER,2,ITJABG,2,IFLNRD,2,0,-18)
3320 CALL MOVCP(1BFSBR,1,2,44,2,IBFEPI,40)
3330 KONTRL=1 ! Initialisation DWPVR
3340 CALL DWPVR(DTENRR,KONTRL,IBFSBR,NTHEPI,0,DTENRR) ! Write TVR epi
3350 IF(KONTRL.EQ.0)GOTO 310 ! Controle write TVR
3360 CALL DBERR(INTT(DTENRR),2,NTHEPI,1,0,2,KONTRL,BBUF,N)
3370 CALL ERRIN('MAIN.',BBUF,4,1) ! Message erreur
3380 GOTO 50
3390 C
3400 C *****
3410 C
3420 C+ Update thesaurus journalier
3430 C
3440 300 LDBL=.TRUE.
3450 310 N=0
```

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 16:50 BLAD 8
EPIOJC

```
3460 320 CALL MOVCP( IBUF, ITHJNR, 56 )      ! Mov OLD pour update
3470      IF( .NOT. LDBL ) CALL SET( 7, ITJBTG ) ! Set bits 1, 2 et 8 (1-16)
3480      IF( LDBL ) ITJBTG = -1              ! Set tous les bits (1-16)
3490      ITJNRD = IFLNRD                     ! Set flag mere ds journalier
3500      CALL DUPTHE( NTHJNR, KONTRL, ITHJNR, IBUF, DTJNRR ) ! Update record ITHJ
3510      IF( KONTRL.EQ.0 ) GOTO 340           ! Test si update Ok
3520      IF( KONTRL.NE.-3.OR.N.GT.3 ) GOTO 330 ! Test si err. multi-users
3530      N = N + 1                           ! Increment. compt. multi-use
3540      GOTO 320
3550 330 CALL DBERR( INTT( DTJNRR ), 1, NTHJNR, 0, 0, 3, KONTRL, BBUF, N )
3560      CALL ERRIN( '.MAIN.', BBUF, 4, 1 )   ! Erreur qcq
3570      GOTO 50
3590 340 IF( .NOT. LDBL ) CALL MOVCP( IBFCMP, IBFTPJ, 12 )
3600      GOTO 50
3610 C
3620 C++      Cloture et fermeture des buffers
3630 C
3640 350 CALL SAY( 'ATTENTION ! PROGRAM IS STOPPED ( ERROR WRITE REC.EPI ) ' )
3650 360 CALL RWRISP( 6, RDNCND( 1, ILAN ), 48 )
3660 370 CALL RALBUF      ! Cloture des buffers
3670      CALL WEGTMR( RNMTJPJ )      ! Release .T
3680 C
3690 C *****
3691 C *****
3700 C
3710      CALL RETPRG
3720 C
3730 C
3740      END
OKAY
```


M O D U L E : E P I J X . L

```
0010 C
0020 C      ???
0030 C
0040 C *****
0050 C
0060 C      PROGRAMMA                : /EPIJX.L
0070 C      .D-FILE + USERNR       : EPIOJX.D, 23.
0080 C      DOCUMENTATIENR         :
0090 C      AUTEUR                  : Pascale Lefebvre
0100 C      GROEP + PROJECT        : Epidemiologie, lecture sign. germes/"B-
0110 C      CREATIEDATUM            : 19-05-1987
0120 C      GEWIJZIGD DOOR         :
0130 C      DATUM LAATSTE WIJZIGING :
0140 C      AARD LAATSTE WIJZIGING  :
0150 C
0160 C *****
0170 C
0180 C      FONCTIONELE BESCHRIJVING :
0190 C
0200 C      1- Lecture du .T cree et fournit par /EPIJR.L
0210 C      Les records .T sont ceux qui ont ete transportes vers TH.EPI
0220 C      et qui n'ont pas encore ete teste pour patho/cibles.
0230 C      2- Lecture des TVRs signaletiques germes pathogenes/cibles
0240 C      3- Lecture des resultats labo (numeration et si necessaire ABg)
0250 C      4- Contrôles divers si resultats labo. acceptables
0260 C      5- Lecture delai de redondance si res.patho /cible.
0270 C      GEBRUIKSAANWIJZING :
0280 C
0290 C *****
0300 C
0310 C$$$
0320 C$PAG 0
0330 C$FC EPIOJX/K/S
0340 C$ALL P.T,100/ZE
0350 C$LINK /EPIJX.L
0360 CP,KB:/SH<I.T/CC,LAB0,EPI,LAB1,LAB0,LAB1,EXPR
0370 C,LCO,ALO,AL1,BRA,BAS/E
0380 C$MANIPO /EPIJX.L/K
0390 C
0400 C$LOPFOV /EPIJX.L<P.T
0410 CJA
0420 C$PAG 24
0430 C$$$
0440 C
0450 C *****
0460 C
0470 C      BLOCK DATA OVRLAY
0480 C      COMMON /RECOVL/          ! Common systeme I/O et overlay
0490 C      COMMON /MENU /           ! Informations generales JOB/BRU
0500 C      COMMON /EBFPRM/         ! Informations parametre epi.
0510 C      COMMON /EBFDTA/         ! Info datas edition
0520 C
0530 C      COMMON /EBFTPJ/          ! Gestion fichier temporaire fich. journalier
0540 C      COMMON /EBFTPE/         ! Gestion fichier temporaire fich. edition
0550 C      COMMON /ETHJNR/         ! Record thes. journalier
```



```
0560      COMMON /EWKIPC/      ! Informations generales EPI
0570 C
0580 C
0590      COMMON /PERS  /      ! Record patient
0600      COMMON /PVR   /      ! Suite record patient
0610      COMMON /LTHDEM/      ! Record demande labo.
0620      COMMON /LBFAND/      ! Suite record demande
0630      COMMON /LBF2AD/      ! Idem.
0640      END
```

[illegible]


```

1210      DATA IDTTPC /2,3/
1220      DATA NTHJNR,NTHBYS /126,115/
1230 C
1240 C -----
1250 C
1260      IF(IVATPJ.NE.1)GOTO 10                ! Test si premier passage
1270      RNMTPE='B.T '                        ! Nom fichier .T
1280      ILUTPE=7                              ! Numero logique
1290      IMXTPE=0                              ! Nbr. max. rec. (0=auto)
1300      ILNTPJ=80                            ! Long. byte record .T
1310      IVATPE=1                              ! Init. var. ass. .T edition
1320      CALL SETTMR(RNMTPE,500)              ! Attribut. .T edition
1330 C
1340      CALL RALOV(ILUTPJ,IBOTPJ,1)           ! Allocation buffer .T JNR
1350      CALL RALOV(ILUTPE,IBOTPE,1)           ! " " " edition
1360      CALL ROPER(ILUTPJ,RNMTPE,IMXTPE,ILNTPJ,IVATPJ) ! Buffer .T JNR
1370      CALL ROPER(ILUTPE,RNMTPE,IMXTPE,ILNTPJ,IVATPE) ! " " edition
1380 C
1390      CALL MOVCPD(DBFCPC,0,-8)              ! Init. zone comparaison
1400 C
1410      CALL RWRISP(6,RDEDIT(1,ILAN),64)
1420 C
1430 C *****
1440 C *****
1450 C
1460      10 IF(IVATPJ.GT.IMXTPE)GOTO 350        ! Test si fin .T JNR
1470      CALL RREAR(ILUTPJ,IBFTPJ,ILNTPJ,IVATPJ)
1480 C
1490      CALL LRDJNR(DBFJNR,56,1,KONTRL)
1500      IF(KONTRL.NE.0)GOTO 10
1510 C
1520 C *****
1530 C
1540 C++      Description .T
1550 C
1560 C      Libelle                               Posit.   Length
1570 C                               byte      byte
1580 C      -----
1590 C      Pointeur B-S "master"             1         4
1600 C      Pointeur Germe                     5         4
1610 C      Reserve                           9        48
1620 C      Numero rec.JNR                    57         4
1630 C
1640 C -----
1650 C
1660      20 IF(IEQUAL(ITHJNR,DBFCPC,8).EQ.0)GOTO 80
1670 C
1680 C++      Test si pathogene et/ou cible
1690 C      Lecture des informations TVR de ce germe dans ce body-site
1700 C      1. Voir dans TVR pathologique (TVR 115.2)
1710 C      2. Voir dans TVR cibles          (TVR 115.3)
1720 C
1730      CALL MOVCPD(IBFGPC,0,-72)
1740      CALL MOVCPD(IBFAPC,0,-224)
1750      DO 60 ID0=1,2
1760      KONTRL=1

```



```

1770      NSKIP=0
1780      ITYP=IDTTTPC(IDO)
1790      CALL DRPVR3(DBFBYS,KONTRL,IBFSBR,2048,NSKIP,ITYP,DTJGER,4,0
1800      *      ,NTHBYS,0,DBFBYS)
1810      IF(KONTRL)30,60,40
1820 30      IF(KONTRL.EQ.-4)GOTO 60
1830      CALL DBERR(INTT(DBFBYS),1,NTHBYS,0,0,1,KONTRL,BBUF,N)
1840      CALL ERRIN('MAIN.',BBUF,4,1)
1850      GOTO 10
1860 40      IF(KONTRL.EQ.1)GOTO 50
1870      N=IEDIT(BBUF)+ITB(49)+IST('MULTIPLE SUBSUBRECORDS "BODY-SITE"')
1880      *      +IIA(6,DBFBYS)+ITB(141)
1890      CALL ERRIN('MAIN.',BBUF,4,1)
1900 50      CALL MOVCP(1,IBFGPC(1,IDO),IBFSBR(3),36)
1910      IF(IBFSBR(2).EQ.40)GOTO 60
1920      CALL MOVCP(1,IBFAPC(1,1,IDO),IBFSBR(21),IBFSBR(20)-4)
1930      NBFAPC(IDO)=(IBFSBR(20)-4)/4
1940 60      CONTINUE
1950      IBTTTPC=0
1960      DO 70 IDO=1,2
1970      IF(ITTEST(1,IBFGPC(1,IDO)).NE.0)CALL SET(IDO-1,IBTTTPC)
1980 70      CONTINUE
1990      CALL MOVCP(1,DBFCPC,ITHJNR,8)
2000 80      IF(IBTTTPC.EQ.0)GOTO 260
2010 C
2020 C *****
2030 C
2040 C++      Lecture des resultats de la demande
2050 C
2060      DNP=DTJPAT      ! Attribution num. pat. DI
2070      CALL LINPIR(DTJPAT,NRPAT,KONTRL)      ! " " " AZL
2080      CALL MOVCP(1,IBFSLC,0,2,0,2,INTT(DTJNTR),2,INTT(DTJNRD),2,0,-8)
2090      CALL MOVCP(1,IBFANA,ITJANA(2),4)      ! Mov analyses culture/ABg
2100      NBRANA=1      ! 1 analyse : culture
2110      DO 90 IDO=1,2      ! Cycle pour voir si ABg
2120      IF(1,IBFAPC(1,1,IDO).NE.0)NBRANA=2      ! oui, 2ieme analyse : ABg
2130 90      CONTINUE
2140      KTLRES=0      ! Init. LRDRES
2150 100 CALL LRDRES(1,IBFSLC,1,3,4,4,IBFANA,NBRANA,IBFSBR,KTLRES)
2160      IF(KTLRES)110,190,120      ! <0:err.,=0:fin, >0:res.
2170 110 CALL SAY('ERROR LRDRES !')
2180      GOTO 10
2190 C
2200 C * * * * *
2210 C
2220 120 CALL PLEXIF(1,FASTT,KONTRL,7,9,10,11) ! Test le statut analyse
2230      IF(KONTRL.GT.4)GOTO 100      ! si diff., pas de resultat
2240      CALL NZOEK(1,IBFANA,2,IBFANA,NBRANA,IPTANA) ! Pour quelle analyse
2250      GOTO(130,160,100),IPTANA      ! Culture ou ABg
2260 C
2270 C+      Mov resultat de la culture
2280 C
2290 130 ILNBAC=MOVCP(1,IBFBAC,IBFSBR(9),IBFSBR(2))
2300      LFLNIS=.FALSE.      ! Flag. culture generale
2310      CALL ONZOEK(0,1,DBBNIS,8,KONTRL)      ! Test si presence cult. gen.
2320      IF(KONTRL.LE.8)LFLNIS=.TRUE.      ! oui, flag est vrai

```


<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 17:44 BLAD 6
EPIOJX

```

2330      NBRGER=(ILNBAC/2-4-12)/12      ! Nombre de germes (=souches)
2340      DO 140 IDO=1,NBRGER            ! Cycle sur les germes
2350      IF(IEQUAL(DTJGER,IBFGER(11,IDO),4).EQ.0)GOTO 150
2360 140  CONTINUE                        ! Test si retrouve meme germe
2370      GOTO 260                        ! non, pas path. ni cible
2380 150  IPTGER=IDO                      ! Germe retrouve, num. souche
2390      GOTO 100                        ! Retour pour lire ABg
2400  C
2410  C      Mov resultat de l'antibiogramme
2420  C
2430 160  ILNBAB=MOVCPL(IBFBAB,IBFSBR(9),IBFSBR(2))
2440      NBRABT=0                        ! Init. nombre ABt/ABg-dem.
2450      IPTBAB=5                        ! " point. zone ABg
2460 170  IF(IPTBAB.GT.ILNBAB)GOTO 100    ! Test si fin de la zone
2470      CALL MOVCPL(IBFRAB,IBFBAB(IPTBAB),IBFBAB(IPTBAB+1))
2480      IF(NUMSCH.EQ.IPTGER)GOTO 180    ! Mov D/work, test meme souche
2490      IPTBAB=IPTBAB + ILNRAB/2        ! non, incremt. point. zone
2500      GOTO 170                        !
2510 180  NBRABT=(ILNRAB-8)/4            ! meme souche D/ABg-dem.
2520  C
2530  C *****
2540  C
2550  C+   Comparaison entre les numerations
2560  C
2570 190  IBTRPC=0                        ! Init. bitst. res. P/C
2580      IBIT=-1                        ! Init. subr. NEXTON
2590 200  CALL NEXTON(IBTRPC,2,IBIT)      ! Cycle sur P/C
2600      IF(IBIT.EQ.-1)GOTO 250          ! Test si fin NEXTON
2610  C
2620      IF(.NOT.LFLNIS)GOTO 210          ! Test autre num. generale
2630      IF(ITTEST(TSUB(DBBNGI,IBFGPC(6,IBIT+1))).LT.0)GOTO 200
2640      IF(ITTEST(TSUB(DBBNGS,IBFGPC(8,IBIT+1))).GT.0)GOTO 200
2650      GOTO 220                          ! Numeration gen. acceptee
2660 210  IF(ITTEST(TSUB(IBFGER(7,IPTGER),IBFGPC(6,IBIT+1))).LT.0)GOTO 200
2670      IF(ITTEST(TSUB(IBFGER(9,IPTGER),IBFGPC(8,IBIT+1))).GT.0)GOTO 200
2671  C                                  ! Numeration P/souche acceptee
2672  C+   Comparaison entre les antibiotiques
2680  C
2690 220  IF(IBFAPC(1,1,IBIT+1).EQ.0)GOTO 240 ! Test si au moins 1 ABt
2700      IF(NBRABT.EQ.0)GOTO 200          ! Refuse, si aucun ABt-dem.
2710      DO 230 IDO=1,NBRABT            ! Cycle sur les ABt
2720      CALL NZOEK(IBFRSH(1,IDO),4,IBFAPC(1,1,IBIT+1),NBFAPC(IBIT+1)
2730      * ,KONTRL)
2740      IF(KONTRL.GT.NBFAPC(IBIT+1))GOTO 230 ! Test ABt+res retrouve
2750      NBRRAB=NBRRAB + 1                ! oui, increment. nbr. ABt
2760      IF(NBRRAB.GE.(NBFAPC(IBIT+1)*IBPPAB)/100)GOTO 240
2770 230  CONTINUE                        ! Test si depasse coeff. acc.
2780      GOTO 200
2790 240  CALL SET(IBIT,IBTRPC)            ! Accepte path. et/ou cible
2800      GOTO 200
2810 250  IF(IBTRPC.NE.0)GOTO 290          ! Test si path. cible accept.
2820  C
2830  C *****
2840  C
2841  C++   Update journalier pour les cas non acceptes
2842  C

```



```

2850      260 N=0                                ! Init. compt. multi-users
2860      270 CALL MOVCP( IBUF, ITHJNR, 56 )      ! mov OLD/update
2870          CALL SET( 0, ITJBTG )              ! Set flag path. termine
2880          CALL SET( 1, ITJBTG )              ! " " cible termine
2890          CALL DUPTHE( NTHJNR, KONTRL, ITHJNR, IBUF, DTJNRR )
2900          IF( KONTRL.EQ.0 ) GOTO 10            ! Update et test si Ok
2910          IF( KONTRL.NE.-3.OR.N.GT.3 ) GOTO 280 ! Test si update multi-users
2920          CALL LRDJNR( DTJNRR, 60, 1, KONTRL ) ! Relecture journalier
2930          N=N + 1                             ! Incremtn. compt. multi-users
2940          GOTO 270
2950      280 CALL DBERR( INTT( DTJNRR ), 1, NTHJNR, 0, 0, 3, KONTRL, BBUF, N )
2960          CALL ERRIN( '.MAIN.', BBUF, 4, 1 )   ! Err. qcq update
2970          GOTO 10
2980  C
2990  C *****
3000  C
3010  C++   Lecture du delai de non redondance pour ce germe dans ce body-site
3020  C     1. Voir dans TVR particularite germe ds B-S (TVR 115.1) germe > 0
3030  C     2. Voir dans TVR generalite body-site (TVR 115.1) germe = 0
3040  C
3050      290 CALL MOVCP( DTTBDG, DBFGER, 4, 0, -4 ) ! Germe specifique/general
3060          DO 330 IDO=1,2                         ! Cycle spec. (#0), gen.=0
3070          KONTRL=1                               ! init. DRPVR3
3080          NSKIP=0                                ! " "
3090          CALL DRPVR3( DBFBYS, KONTRL, IBFSBR, 2048, NSKIP, 1, DTTBDG( IDO ), 4, 0
3100          * , NTHBYS, 0, DBFBYS )
3110          IF( KONTRL ) 300, 330, 310              ! test si retrouve
3120      300 IF( KONTRL.EQ.-4 ) GOTO 330             ! Test si jamais ecrit
3130          CALL DBERR( INTT( DTJBYS ), 2, NTHBYS, 0, 0, 1, KONTRL, BBUF, N )
3140          CALL ERRIN( '.MAIN.', BBUF, 4, 1 )     ! Err. qcq lecture
3150          GOTO 330
3160      310 IF( KONTRL.EQ.1 ) GOTO 320              ! Test si un seul subsubrec.
3170          N=IEDIT( BBUF ) + ITB( 49 ) + IST( 'MULTIPLE SUBSUBRECORD "BODY-SITE"' )
3180          * + IAA( 6, DBFTPJ( 1 ) ) + ITB( 141 )
3190          CALL ERRIN( '.MAIN.', BBUF, 4, 1 )     ! Multi subsubrecords
3200          IDLRDD=IBFSBR( 6 )                     ! Delai de redondance
3210      320 GOTO 340
3220      330 CONTINUE                               ! Non retrouve spec./gen.
3230          IDLRDD=IDLRDG                          ! Delai de redondance param.
3240      340 CALL RUN( '/EPIJY.L' )                 ! Run vers mod. test result.
3250  C
3260  C *****
3270  C *****
3280  C *****
3290  C
3300      350 CALL RUN( '/EPIJL.L' )                ! Run vers mod. edition
3310  C
3320  C
3330      END
OKAY

```


M O D U L E : E P I J Y . L

```

0001 C
0002 C      ???
0003 C
0004 C *****
0005 C
0006 C      PROGRAMMA                : /EPIJY.L
0007 C      .D-FILE + USERNR        : EPI0JY.D, 23.
0008 C      DOCUMENTATIENR          :
0009 C      AUTEUR                    : Pascale Lefebvre
0010 C      GROEP + PROJECT           : Epidemiologie, gestion res. stat. epi.
0011 C      CREATIEDATUM              : 19-05-1987
0012 C      GEWIJZIGD DOOR           :
0013 C      DATUM LAATSTE WIJZIGING   :
0014 C      AARD LAATSTE WIJZIGING   :
0015 C
0016 C *****
0017 C
0018 C      FUNCTIONELE BESCHRIJVING :
0019 C
0020 C++      Test si encore hospitalise maintenant
0021 C
0022 C      A partir du record journalier accepte (patho ou cible) et
0023 C      connaissant le delai de non redondance de ce germe,
0024 C      ce module ecrit dans le .T (RNMTPJ) l'occurrence a editer.
0025 C      Si le delai germe <> delai rec.epi ==> CALL CLORUN ('/EPIJZ.L')
0026 C
0027 C      Description du .T d'edition
0028 C      -----
0029 C      libelle                     POSIT.    POSIT.    LENGTH
0030 C                                MOT      BYTES
0031 C
0032 C      Pointeur B-5                 1         1         4
0033 C      Pointeur germe               3         5         4
0034 C      Pointeur patient JNR         5         9         4
0035 C      Numero admission JNR         7        13         4
0036 C      Numero demande JNR          9        17         8
0037 C      Pointeur serv.inf JNR       13        25         2
0038 C      Pointeur chambre JNR       14        27         2
0039 C      Ref.liste JNR               15        29         2      1=P/2=C
0040 C      Num.analyse B-5 JNR         16        31         2
0041 C      Num.analyse CLT JNR         17        33         2
0042 C      Num.analyse ABg JNR         18        35         2
0043 C      Num pat.contaminant          19        37         4
0044 C      Num adm contaminant          21        41         4
0045 C      Serv.                        23        45         2
0046 C      Chbre                        24        47         2
0047 C      Contaminant deja edite =*    25        49         2      0=non;1=o
0048 C      Flag redondant (IFLRED)     26        51         2      1=red/0=N
0049 C      Date dem.contaminant         27        53         2
0050 C      Numero dem.contaminant       28        55         8
0051 C      Date admission JRN           32        63         2
0052 C      Date demande JRN            33        65         2
0053 C      reserve                      34        67        14
0054 C      GEBRUIKSAANWIJZING          :
0055 C      Programme EPISTA.

```



```

0048 C
0049 C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
0050 C $$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$$
0051 C
0052 PROGRAM EPIOJD
0053 C
0054 C
0055 COMMON /RECOVL/IRCOVL(65)
0056 COMMON /MENU /ISTR(80)
0057 C
0058 COMMON /ETHJNR/ITHJNR(30)
0059 C
0060 REAL*8 RDEDIT(8,2)
0061 DIMENSION IBUF(256)
0062 BYTE BBUF(144)
0063 C
0064 EQUIVALENCE (ISTR(55),ILAN)
0065 EQUIVALENCE (ITHJNR(7),ITJB TG),(ITHJNR(29),DTJNRR)
0066 EQUIVALENCE (IBUF(1),BBUF(1))
0067 C
0068 DATA RDEDIT /'Mise a j','our des ','resultat','s "journ","aliers" '
0069 *,': pathog','enes et ','cibles& ','Wijzigin','g van de',' "dageli'
0070 *,'jkse" r','esultate','n : patho','genen en',' doelen&'/'
0071 C
0072 DATA NTHJNR /126/
0073 C
0074 C -----
0075 C
0076 CALL MPARAM ! Reception module SUBPRG
0077 C
0078 C *****
0079 C *****
0080 C
0081 CALL RWRISP(6,RDEDIT(1,ILAN),64) ! Write message controle
0082 C
0083 C++ Initialisations
0084 C
0085 CALL RALBUF(8,-1) ! Attribution du buffer
0086 CALL ROPETH(8,NTHJNR) ! Relation num. log./th. jour
0087 C
0088 C++ Lecture du journalier
0089 C
0090 10 CALL RREATB(8,ITHJNR,28,DTJNRR,KONTRL) ! ctur e th es . journalier
0091 IF(KONTRL.EQ.-1)GOTO 20 ! Test si fin de fichier
0092 IF((ITJB TG.AND.131).NE.131)GOTO 10 ! Test si deja "move"
0093 C
0094 CALL MOV CPL(IBUF,ITHJNR,28) ! Mov OLD/update
0095 CALL DELTHE(NTHJNR,KONTRL,ITHJNR,IBUF,DTJNRR) ! Delete record journ.
0096 IF(KONTRL.EQ.0)GOTO 10 ! Test si update Ok
0097 CALL DBERR(INTT(DTJNRR),1,NTHJNR,0,0,5,KONTRL,BBUF,N)
0098 CALL ERRIN('MAIN.',BBUF,4,1) ! Erreur qcq
0099 GOTO 10
0100 20 CALL RALBUF ! Fermeture des buffers
0101 C
0102 C *****
0103 C *****

```

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 18:09 BLAD 3
EPIOJD

0104 C

0105 CALL RETPRG

! Retour au mod. appellant

0106 C

0107 C

0108 END

OKAY

M O D U L E : L R D B Y S

<<SRC>> VOOR USER 37 TYDENS: TESTRU-KSJ K1 11-09-87 11:45 BLAD 1
LRDBYS

```
0001 C
0002 C      ???
0003 C
0004 C *****
0005 C
0006 C      SUBROUTINE                : LRDBYS
0007 C      .D-FILE + USERNR         : LRDBYS.D, 37.
0008 C      DOCUMENTATIENR           :
0009 C      AUTEUR                    : Pascale Lefebvre
0010 C      GROEP + PROJECT           : LABO, bacterio.
0011 C      CREATIEDATUM              : 27-04-1987
0012 C      GEWIJZIGD DOOR           :
0013 C      DATUM LAATSTE WIJZIGING   :
0014 C      AARD LAATSTE WIJZIGING   :
0015 C
0016 C *****
0017 C
0018 C $$
0019 C $PAG 0
0020 C $FC LRDBYS/K/S
0021 C $LIBR
0022 C LABO<LABO,I/R
0023 C YE
0024 C $PAG 24
0025 C $$
0026 C
0027 C *****
0028 C
0029 C      FUNCTIONELE BESCHRIJVING :
0030 C      Lecture record thes. "body-site" a partir d'un pointeur DI
0031 C
0032 C      FORTRAN INTERFACE        :
0033 C      CALL LRDBYS(DPTBYS,ILNREC,IOFREC,KONTRL)
0034 C      *PARAMETERLIJST
0035 C      IN : DPTBYS : D*4                : Numero de record DI du record
0036 C                                           : "Body-syte" a lire
0037 C      ILNREC : I*2                    : Longueur en bytes des infos. a
0038 C                                           : "mover" dans la zone ITHBYS
0039 C      IOFREC : I*2                    : Byte offset (a partir de) a
0040 C                                           : "mover" dans la zone ITHBYS
0041 C      UIT: KONTRL : I*2                : Parametre de controle
0042 C                                           : 0 = ok
0043 C                                           : #0= erreur qcq de lecture
0044 C      *COMMON LTHBYS
0045 C      UIT: ITHBYS : I*144              : Zone thesaurus lu
0046 C
0047 C *****
0048 C
0049 C      SUBROUTINE LRDBYS(DPTBYS,ILNREC, C
0051 C      COMMON /LTHBYS/ITHBYS(72)
0052 C
0053 C      BYTE BBUF(144)
0054 C
0055 C      EQUIVALENCE (ITHBYS(71),DTYNRR)
```


<<SRC>> VOOR USER 37 TYDENS: TESTRU-KSJ K1 11-09-87 11:45 BLAD 2
LRDBYS

```
0056          DATA NTHBYS /115/
0057 C
0058 C -----
0059 C
0060          CALL DRETHER(NTHBYS,KONTRL,ITHBYS,DPTBYS,ILNREC,IOFREC)
0061          IF(KONTRL.EQ.0)GOTO 10                ! Test si lecture Ok
0062          CALL DBERR(INTT(DPTBYS),1,NTHBYS,0,0,1,KONTRL,BBUF,N)
0063          CALL ERRIN('LRDBYS',BBUF,4,1)         ! Erreur qcq lecture
0064          GOTO 20
0065          10 DTYNRR=DPTBYS                        ! Attribution numero record
0066 C
0067          20 RETURN
0068 C
0069 C
0070          END
OKAY
```

M O D U L E : L R D G E R

<<SRC>> VOOR USER 37 TYDENS: TESTRU-KSJ K1 11-09-87 11:45 BLAD 1
LRDGER

```
0001 C
0002 C      ???
0003 C
0004 C *****
0005 C
0006 C      SUBROUTINE                : LRDGER
0007 C      .D-FILE + USERNR          : LRDGER.D, 37.
0008 C      DOCUMENTATIENR             :
0009 C      AUTEUR                      : Pascale Lefebvre
0010 C      GROEP + PROJECT             : LABO, bacteriologie
0011 C      CREATIEDATUM                : 27-04-1987
0012 C      GEWIJZIGD DOOR             :
0013 C      DATUM LAATSTE WIJZIGING     :
0014 C      AARD LAATSTE WIJZIGING      :
0015 C
0016 C *****
0017 C
0018 C      FUNCTIONELE BESCHRIJVING :
0019 C      Lecture record des germes a partir d'un pointeur DI
0020 C
0021 C      FORTRAN INTERFACE          :
0022 C      CALL LRDGER(DPTGER,ILNREC,IOFREC,KONTRL)
0023 C      *PARAMETERLIJST
0024 C      IN : IPTGER : D*4           : Numero de record DI du record
0025 C                                           germe a lire
0026 C      ILNREC : I*2               : Longueur en bytes des infos a
0027 C                                           "mover" dans la zone ITHGER
0028 C      IOFREC : I*2               : Byte offset (a partir de) a
0029 C                                           "mover" dans la zone ITHGER
0030 C      UIT: KONTRL : I*2           : Parametre de controle
0031 C                                           0 = ok
0032 C                                           #0= erreur qcq de lecture
0033 C      *COMMON LTHGER
0034 C      UIT: ITHGER : I*184        :
0035 C
0036 C *****
0037 C
0038 C$$
0039 C$PAG 0
0040 C$FC LRDGER/K/5
0041 C$LIBR
0042 CLABO<LABO,I/R
0043 CYE
0044 C$PAG 24
0045 C$$
0046 C
0047 C *****
0048 C
0049 C      SUBROUTINE LRDGER(DPTGER,ILNREC,IOFREC,KONTRL)
0050 C
0051 C      COMMON /LTHGER/ITHGER(92)
0052 C
0053 C      BYTE BBUF(144)
0054 C
0055 C      EQUIVALENCE (ITHGER(91),DTGNRR)
```

<<SRC>> VOOR USER 37 TYDENS: TESTRU-KSJ K1 11-09-87 11:45 BLAD 2
LRDGER

```
0056      EQUIVALENCE (ITHGER(1),BBUF(1))
0057      DATA NTHGER /113/
0058  C
0059  C -----
0060  C
0061      CALL DRETHER(NTHGER,KONTRL,ITHGER,DPTGER,ILNREC,IOFREC)
0062      IF(KONTRL.EQ.0)GOTO 10          ! Test si lecture Ok
0063      CALL DBERR(INTT(DPTGER),1,NTHGER,0,0,1,KONTRL,BBUF,N)
0064      CALL ERRIN('LRDGER',BBUF,4,1)  ! Erreur qcq lecture
0065      GOTO 20
0066  10 DTGNRR=DPTGER                  ! Attributoin numero record
0067  C
0068  20 RETURN
0069  C
0070  C
0071      END
OKAY
```


M O D U L E : L R D I P E

LRDIPE

```

0001 C
0002 C      ???
0003 C
0004 C *****
0005 C
0006 C      SUBROUTINE                : LRDIPE
0007 C      .D-FILE + USERNR        : LRDIPE.D, 23.
0008 C      DOCUMENTATIENR           :
0009 C      AUTEUR                    : Pascale Lefebvre
0010 C      GROEP + PROJECT           : Epidemiologie
0011 C      CREATIEDATUM              : 27-04-87
0012 C      GEWIJZIGD DOOR           :
0013 C      DATUM LAATSTE WIJZIGING   :
0014 C      AARD LAATSTE WIJZIGING   :
0015 C
0016 C *****
0017 C
0018 C      FUNCTIONELE BESCHRIJVING :
0019 C
0020 C      Lecture records "data" epidemiologie.
0021 C
0022 C      FORTRAN INTERFACE        :
0023 C
0024 C      CALL LRDIPE(IMINDT,IMAXDT,IMDTRI,BBFTRI,ILNTRI,IBFSBR,ILNSBR
0025 C      *,KONTRL)
0026 C
0027 C      *PARAMETERLIJST :
0028 C
0029 C      IN : IMINDT : I*2                : Date minimum lin. pour selection
0030 C           IMAXDT : I*2                :      " maximum      "      "
0031 C           IMDTRI : I*2                : Mode de tri : 0 = pas de tri
0032 C           BBFTRI : ILNTRI*1           : Zone byte ascii avec la clef
0033 C           ILNTRI : I*2                : Longueur en byte (B*1) de la zone
0034 C                                     : BBFTRI
0035 C           IBFSBR : ILNSBR*2           : Zone scratch de lecture
0036 C           ILNSBR : I*2                : Longueur en mot (I*2) de la zone
0037 C                                     : IBFSBR
0038 C      UIT : KONTRL : INTEGER           : Parametre de controle
0039 C                                           0      = Read O.K.
0040 C                                           <> 0    = Erreur
0041 C
0042 C      *COMMON /ETHEPI/
0043 C      UIT: ITHEPI : I*18
0044 C      *COMMON /EBFTPF/
0045 C      IN : RNMTPF : R*4                : Nom du fichier temporaire
0046 C           ILUTPF : I*2                : Logical unit du fichier temporaire
0047 C           IMXTPF : I*2                : Nbre max.de records du fichier tem
0048 C           ILNTPF : I*2                : Longueur du record fich. temp.
0049 C           IVATPF : I*2                : Variable associee du fichier temp.
0050 C      UIT: IDEM :
0051 C
0052 C *****
0053 C
0054 C $$$
0055 C $PAG 0

```


<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 15:23 BLAD 2
LRDIPE

```

0056 C$FC LRDIPE/K/5
0057 C$LIBR
0058 CEPI<EPI,I.T/R
0059 CYE
0060 C$PAG 24
0061 C$$
0062 C
0063 C *****
0064 C
0065 SUBROUTINE LRDIPE(IMINDT,IMAXDT,IMDTRI,BBFTRI,ILNTRI,IBFSBR,ILNSBR
0066 *,KONTRL)
0067 C
0068 COMMON /ETHEPI/ITHEPI(18)
0069 COMMON /EBFTPF/RNMTPF,ILUTPF,IMXTPF,ILNTPF,IVATPF
0070 C
0071 BYTE BBFTRI(ILNTRI),BBFEDT(132),BBUF(144)
0072 DIMENSION IBFSBR(ILNSBR),IBFBLK(40),IRP(2)
0073 C
0074 EQUIVALENCE (ITHEPI(17),DTENRR)
0075 EQUIVALENCE (IBFBLK(3),IDMDAT)
0076 C
0077 C -----
0078 C
0079 NSKIP=0 ! Init. NSKIP DRPVR2
0080 10 CALL DRPVR2(DTENRR,KONTRL,IBFSBR,ILNSBR*2,NSKIP,0,0,TYP,0,NTHEPI)
0081 IF(KONTRL)20,60,30 ! <0=err., 0=fin(?), >0=N sub
0082 20 IF(KONTRL.EQ.-4)GOTO 60 ! Aucun record trouve
0083 CALL DBERR(INTT(DTENRR),1,NTHEPI,0,0,1,KONTRL,BBUF,N)
0084 CALL ERRIN('LRDIPE',BBUF,4,1) ! Erreur qcq lecture
0085 GOTO 90
0086 30 NRSBR=KONTRL ! Nombre de subsubrecords lus
0087 C
0088 C *****
0089 C
0090 IPTSBR=1 ! Init. point (mot) IBFSBR
0091 DO 50 IDO=1,NRSBR ! Cycle sur les N subsubr.
0092 ILNBLK=IBFSBR(IPTSBR+1) ! Longueur du subsubrecord
0093 CALL MOVCP(LIBFBLK,IBFSBR(IPTSBR),ILNSBR) ! Mov ds zone "work"
0094 IF(IDMDAT.LT.IMINDT)GOTO 40 ! Test si < date minimum
0095 IF(IDMDAT.GT.IMAXDT)GOTO 40 ! " " > " maximum
0096 CALL RWRIRS(ILUTPF,IBFBLK,ILNTPF,IVATPF) ! Accepte, write .T
0097 40 IPTSBR=IPTSBR+ILNBLK/2 ! Incrementation point. subs.
0098 50 CONTINUE
0099 60 IF(NSKIP.GT.0) GOTO 10 ! Test si encore subs. a lire
0100 IMXTPF=IVATPF - 1 ! Nombre de records .T
0101 C
0102 IF(IMDTRI.EQ.0)GOTO 80 ! Test si demande de tri
0103 C
0104 C *****
0105 C
0106 CALL MOVCP(LIBFBLK,'0',-ILNTPF) ! EOF
0107 CALL RSORT(ILUTPF,RNMTPF,BBFTRI,IRP) ! Write .T EOF
0108 IF(IRP(1).EQ.0) GOTO 70 ! Test si tri Ok
0109 CALL DBERR(0,0,0,0,0,6,KONTRL,BBUF,N) ! Erreur qcq de tri
0110 CALL ERRIN('LRDIPE',BBUF,4,1) ! Write err. fichier messages
0111 GOTO 90

```

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 15:23 BLAD 3
LRDIPE

```
0112      70 IMXTPF=IRP(2)                      ! Nombre de records .T
0113 C
0114 C *****
0115 C
0116      80 KONTRL=1                          ! Flag end routine OK
0117 C
0118      90 RETURN
0119 C
0120 C
0121      END
OKAY
```


M O D U L E : LRDEPI

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 15:23 BLAD 1
LRDEPI

```
0010 C
0020 C      ???
0030 C
0040 C *****
0050 C
0060 C      SUBROUTINE                : LRDEPI
0070 C      .D-FILE + USERNR        : LRDEPI.D, 37.
0080 C      DOCUMENTATIENR          :
0090 C      AUTEUR                   : Pascale Lefebvre
0100 C      GROEP + PROJECT          : Epidemiologie.
0110 C      CREATIEDATUM              : 27-04-1987
0120 C      GEWIJZIGD DOOR           :
0130 C      DATUM LAATSTE WIJZIGING  :
0140 C      AARD LAATSTE WIJZIGING   :
0150 C
0160 C *****
0170 C
0180 C $$
0190 C $PAG 0
0200 C $FC LRDEPI/K/5
0210 C $LIBR
0220 C CEPI<EPI,I/R
0230 C YE
0240 C $PAG 24
0250 C $$
0260 C
0270 C *****
0280 C
0290 C      FUNCTIONELE BESCHRIJVING :
0300 C      Lecture record thes. des germes a partir d'un pointeur DI
0310 C
0320 C      FORTRAN INTERFACE        :
0330 C      CALL LRDEPI(DPTEPI,ILNREC,IOFREC,KONTRL)
0340 C      *PARAMETERLIJST
0350 C      IN : DPTEPI : D*4          : Numero de record DI du record
0360 C                                     germe a lire
0370 C      ILNREC : I*2              : Longueur en bytes des infos. a
0380 C                                     "mover" dans la zone ITHEPI
0390 C      IOFREC : I*2              : Byte offset (a partir de) a
0400 C                                     "mover" dans la zone ITHEPI
0410 C      UIT: KONTRL : I*2          : Parametre de controle
0420 C                                     0 = ok
0430 C                                     #0= erreur qcq de lecture
0440 C      *COMMON ETHEPI
0450 C      UIT: ITHEPI : I*36         : Zone thesaurus lu
0460 C
0470 C *****
0480 C
0490 C      SUBROUTINE LRDEPI(DPTEPI,ILNREC,IOFREC,KONTRL)
0500 C
0510 C      COMMON /ETHEPI/ITHEPI(18)
0520 C
0530 C      BYTE BBUF(144)
0540 C
0550 C      EQUIVALENCE (ITHEPI(17),DTENRR)
```


<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 15:23 BLAD 2
LRDEPI

```
0570      DATA NTHEPI /103/
0580  C
0590  C -----
0600  C
0620      CALL DRETHER(NTHEPI,KONTRL,ITHEPI,DPTEPI,ILNREC,IOFREC)
0630      IF(KONTRL.EQ.0)GOTO 10          ! Test si lecture Ok
0640      CALL DBERR(INTT(DPTEPI),1,NTHEPI,0,0,1,KONTRL,BBUF,N)
0650      CALL ERRIN('LRDEPI',BBUF,4,1)    ! Erreur qcq lecture
0660      GOTO 20
0670  10 DTENRR=DPTEPI                    ! Attribution numero record
0680  C
0690  20 RETURN
0700  C
0710  C
0720      END
OKAY
```

M O D U L E : L R D J N R

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 15:23 BLAD 1
LRDJNR

```
0010 C
0020 C      ???
0030 C
0040 C *****
0050 C
0060 C      SUBROUTINE                : LRDJNR
0070 C      .D-FILE + USERNR          : LRDJNR.D, 23.
0080 C      DOCUMENTATIENR             :
0090 C      AUTEUR                      : Pascale Lefebvre
0100 C      GROEP + PROJECT            : Epidemiologie.
0110 C      CREATIEDATUM               : 27-04-1987
0120 C      GEWIJZIGD DOOR             :
0130 C      DATUM LAATSTE WIJZIGING    :
0140 C      AARD LAATSTE WIJZIGING     :
0150 C
0160 C *****
0170 C
0180 C$$
0190 C$PAG 0
0200 C$FC LRDJNR/K/S
0210 C$LIBR
0220 CEPI<EPI,I/R
0230 CYE
0240 C$PAG 24
0250 C$$
0260 C
0270 C *****
0280 C
0290 C      FUNCTIONELE BESCHRIJVING :
0300 C      Lecture record thes. journalier epi. a partir d'un pointeur DI
0310 C
0320 C      FORTRAN INTERFACE          :
0330 C      CALL LRDJNR(DPTJNR,ILNREC,IOFREC,KONTRL)
0340 C      *PARAMETERLIJST
0350 C      IN : DPTJNR : D*4          : Numero de record DI du record
0360 C                                     "JNR" a lire
0370 C      ILNREC : I*2              : Longueur en bytes des infos. a
0380 C                                     "mover" dans la zone ITHJNR
0390 C      IOFREC : I*2              : Byte offset (a partir de) a
0400 C                                     "mover" dans la zone ITHJNR
0410 C      UIT: KONTRL : I*2          : Parametre de controle
0420 C                                     0 = ok
0430 C                                     #0= erreur qcq de lecture
0440 C      *COMMON ETHJNR
0450 C      UIT: ITHJNR : I*60         : Zone thesaurus lu
0460 C
0470 C *****
0480 C
0490 C      SUBROUTINE LRDJNR(DPTJNR,ILNREC,IOFREC,KONTRL)
0500 C
0510 C      COMMON /ETHJNR/ITHJNR(30)
0520 C
0530 C      BYTE BBUF(144)
0540 C
0550 C      EQUIVALENCE (ITHJNR(29),DTJNRR)
```

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 15:23 BLAD 2
LRDJNR

```
0560          DATA NTHJNR /126/
0570 C
0580 C -----
0590 C
0600          CALL DRETHER(NTHJNR,KONTRL,ITHJNR,DPTJNR,ILNREC,IOFREC)
0610          IF(KONTRL.EQ.0)GOTO 10          ! Test si lecture Ok
0620          CALL DBERR(INTT(DPTJNR),1,NTHJNR,0,0,1,KONTRL,BBUF,N)
0630          CALL ERRIN('LRDJNR',BBUF,4,1)    ! Erreur qcq lecture
0640          GOTO 20
0650          10 DTJNRR=DPTJNR                  ! Attribution numero record
0660 C
0670          20 RETURN
0680 C
0690 C
0700          END
OKAY
```


M O D U L E : DSRGER

<<SRC>> VOOR USER 37 TYDENS: TESTRU-KSJ K1 11-09-87 11:45 BLAD 1
DSRGER

```
0001 C
0002 C      ???
0003 C
0004 C *****
0005 C
0006 C      SUBROUTINE                : DSRGER
0007 C      .D-FILE + USERNR          : DSRGER.D, 37.
0008 C      DOCUMENTATIENR            :
0009 C      AUTEUR                     : Christian Pauly
0010 C      GROEP + PROJECT            : Bacteriologie
0011 C      CREATIEDATUM               : 02-06-1987
0012 C      GEWIJZIGD DOOR            :
0013 C      DATUM LAATSTE WIJZIGING    :
0014 C      AARD LAATSTE WIJZIGING     :
0015 C
0016 C *****
0017 C
0018 C      FUNCTIONELE BESCHRIJVING :
0019 C
0020 C      Search record fichier germe. On ne permet pas la notion "*"
0021 C
0022 C      FORTRAN INTERFACE          :
0023 C      DI=DSRGER(BBFGGER)
0024 C
0025 C      *PARAMETERLIJST :
0026 C      IN : BBFGGER : B*8                : Zone ascii avec le code du germe
0027 C                                           : a rechercher
0028 C      UIT: DSRGER : DI*4                : Numero de record DI du germe recher-
0029 C                                           : che : <0 = err. qcq
0030 C                                           : 0 = non retrouve
0031 C                                           : >0 = pointeur germe
0032 C
0033 C *****
0034 C
0035 C $$
0036 C $PAG 0
0037 C $FC DSRGER/K/S
0038 C $LIBR
0039 C LAB0<LAB0,I.T/R
0040 C YE
0041 C $PAG 24
0042 C $$
0043 C
0044 C *****
0045 C
0046 C      REAL FUNCTION DSRGER(BBFGGER)
0047 C
0048 C      DIMENSION IBFFLP(4)
0049 C      BYTE BBFGGER(8)
0050 C
0051 C      DATA NTHGER /113/
0052 C
0053 C -----
0054 C
0055 C      CALL MOVCP(LIBFFLP,-1,-8)
```


<<SRC>> VOOR USER 37 TYDENS: TESTRU-KSJ K1 11-09-87 11:45 BLAD 2
DSRGER

```
0056      KONTRL=-1
0057      CALL DSETHE(NTHGER,KONTRL,BBFGER,DSRGER,1,0,IBFFLP,DI)
0058      IF(KONTRL.LE.0)DSRGER=TINT(KONTRL)
0059  C
0060      RETURN
0061  C
0062  C
0063      END
OKAY
```

M O D U L E : D S R E P I

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 15:23 BLAD 1
DSREPI

```
0010 C
0020 C      ???
0030 C
0040 C *****
0050 C
0060 C      SUBROUTINE                : DSREPI (FUNCTION)
0070 C      .D-FILE + USERNR        : DSREPI.D, 23.
0080 C      DOCUMENTATIENR           :
0090 C      AUTEUR                    : Pascale Lefebvre
0100 C      GROEP + PROJECT          : Epidemiologie
0110 C      CREATIEDATUM              : 27-04-87
0120 C      GEWIJZIGD DOOR           :
0130 C      DATUM LAATSTE WIJZIGING   :
0140 C      AARD LAATSTE WIJZIGING    :
0150 C
0160 C *****
0170 C
0180 C      FUNCTIONELE BESCHRIJVING :
0190 C
0200 C      Search record epidemio. a partir des pointeurs : body-site
0210 C                                                         : germe
0220 C                                                         : patient
0230 C      FORTRAN INTERFACE          :
0240 C      DI=DSREPI(DPTBYS,DPTGER,DPTPAT,BBUF)
0250 C
0260 C      *PARAMETERLIJST :
0270 C      IN   : DPTBYS : D*4                : Numero de record DI du record
0280 C                                         : "body-site" a lire.
0290 C      DPTGER : D*4                : Numero de record DI du record
0300 C                                         : "germe" a lire.
0310 C      DPTPAT : D*4                : Numero de record DI du record
0320 C                                         : patient a lire.
0330 C      BBUF   : B*144              : Zone scratch messages erreurs
0340 C      UIT : DSREPI : D*4                : Parametre de controle + res. search
0350 C                                         : >0= pointeur du candidat retrouve
0360 C                                         : 0= Pas de candidat
0370 C                                         : <0= erreur qcq de search
0380 C
0390 C *****
0400 C
0410 C$$$
0420 C$PAG 0
0430 C$FC DSREPI/K/S
0440 C$LIBR
0450 CEPI<EPI,I.T/R
0460 C$YE
0470 C$PAG 24
0480 C$$$
0490 C
0500 C *****
0510 C
0520 C      REAL FUNCTION DSREPI(DPTBYS,DPTGER,DPTPAT,BBUF)
0530 C
0540 C
0550 C      DIMENSION DBFSRC(3),DBFFLP(3)
```

<<SRC>> VOOR USER 23 TYDENS: TESTRU-KSJ K0 11-09-87 15:23 BLAD 2
DSREPI

```
0560      BYTE BBUF(144)
0570  C
0580      DATA NTHEPI /103/
0590  C
0600  C -----
0610  C
0620      CALL MOVCPPL(DBFSRC,DPTBYS,4,DPTGER,4,DPTPAT,4)
0630      CALL MOVCPPL(DBFFLP,-1,-12)
0640      KONTRL=-1
0650      CALL DSETHE(NTHEPI,KONTRL,DBFSRC,DSREPI,1,INDEX,DBFFLP,DI)
0660      IF(KONTRL)10,20,20
0670  10 CALL DBERR(0,1,NTHEPI,0,0,4,KONTRL,BBUF,N)
0680      CALL ERRIN('DSREPI',BBUF,4,1)
0690  C
0700  20 RETURN
0710  C
0720  C
0730      END
OKAY
```


M O D U L E : DSRBYS

<<SRC>> VOOR USER 37 TYDENS: TESTRU-KSJ K1 11-09-87 11:45 BLAD 1
DSRBYS

```
0001 C
0002 C      ???
0003 C
0004 C *****
0005 C
0006 C      SUBROUTINE                : DSRBYS
0007 C      .D-FILE + USERNR          : DSRBYS.D, 37.
0008 C      DOCUMENTATIENR            :
0009 C      AUTEUR                     : Christian Pauly
0010 C      GROEP + PROJECT            : Bacteriologie
0011 C      CREATIEDATUM               : 02-06-1987
0012 C      GEWIJZIGD DOOR            :
0013 C      DATUM LAATSTE WIJZIGING    :
0014 C      AARD LAATSTE WIJZIGING    :
0015 C
0016 C *****
0017 C
0018 C      FUNCTIONELE BESCHRIJVING :
0019 C
0020 C      Search record fichier "B-S". On ne permet pas la notion "*"
0021 C
0022 C      FORTRAN INTERFACE          :
0023 C      DI=DSRBYS(BBFBYS)
0024 C
0025 C      *PARAMETERLIJST :
0026 C      IN : BBFBYS : B*8                : Zone ascii avec le code du "BS"e
0027 C                                           a rechercher
0028 C      UIT: DSRBYS : DI*4                : Numero de record DI du "BS" recher-
0029 C                                           che : <0 = err. qcq
0030 C                                           0 = non retrouve
0031 C                                           >0 = pointeur "BS"
0032 C
0033 C *****
0034 C
0035 C $$
0036 C $PAG 0
0037 C $FC DSRBYS/K/S
0038 C $LIBR
0039 C LAB0<LAB0,I.T/R
0040 C YE
0041 C $PAG 24
0042 C $$
0043 C
0044 C *****
0045 C
0046 C      REAL FUNCTION DSRBYS(BBFBYS)
0047 C
0048 C      DIMENSION IBFFLP(4)
0049 C      BYTE BBFBYS(8)
0050 C
0051 C      DATA NTHBYS /115/
0052 C
0053 C -----
0054 C
0055 C      CALL MOVCP(LIBFFLP,-1,-8)
```

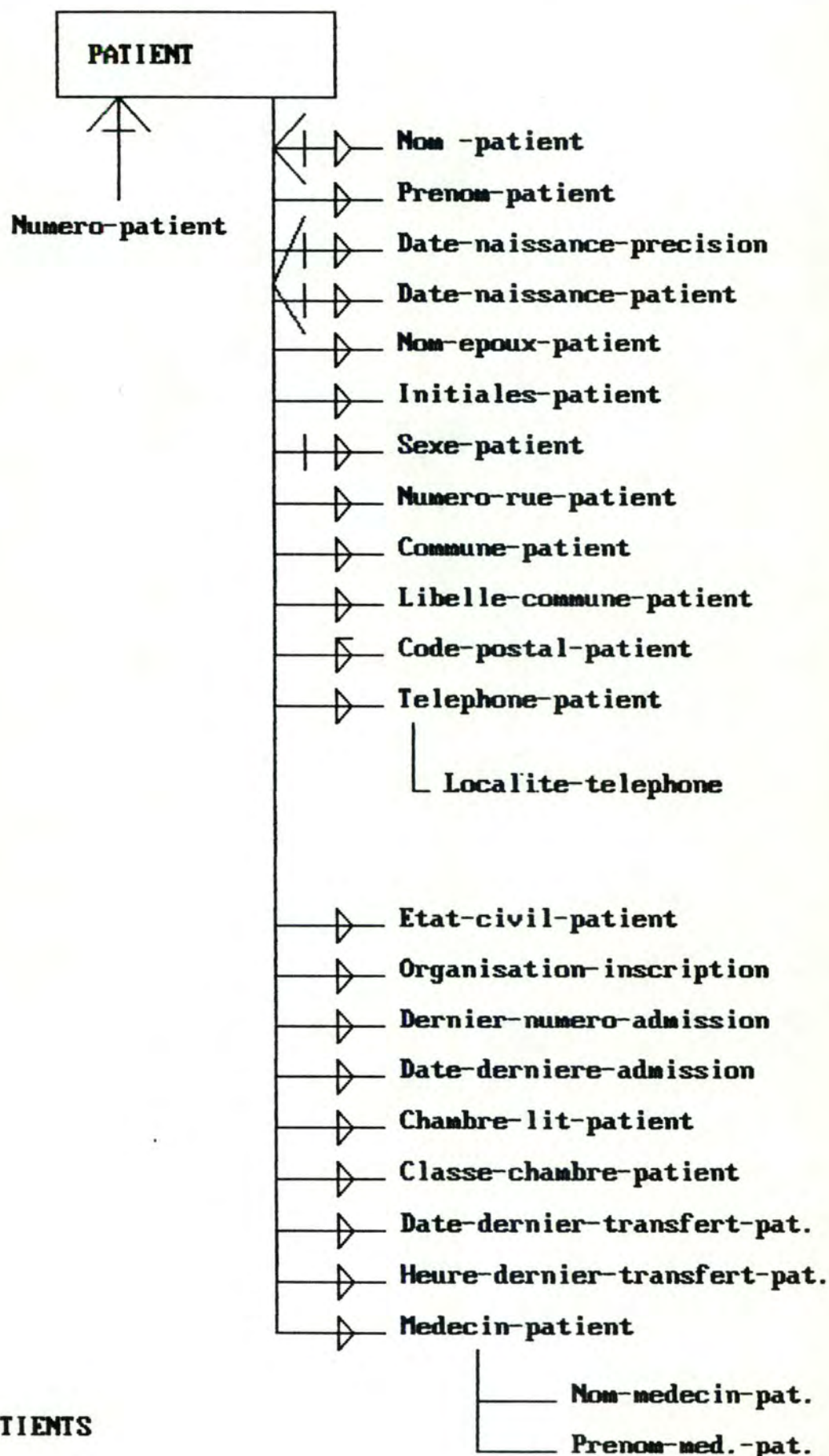

<<SRC>> VOOR USER 37 TYDENS: TESTRU-KSJ K1 11-09-87 11:45 BLAD 2
DSRBYS

```
0056      KONTRL=-1
0057      CALL DSETHE(NTHBYS,KONTRL,BBFBYS,DSRBYS,1,0,IBFFLP,DI)
0058      IF(KONTRL.LE.0)DSRBYS=TINT(KONTRL)
0059  C
0060      RETURN
0061  C
0062  C
0063      END
OKAY
```

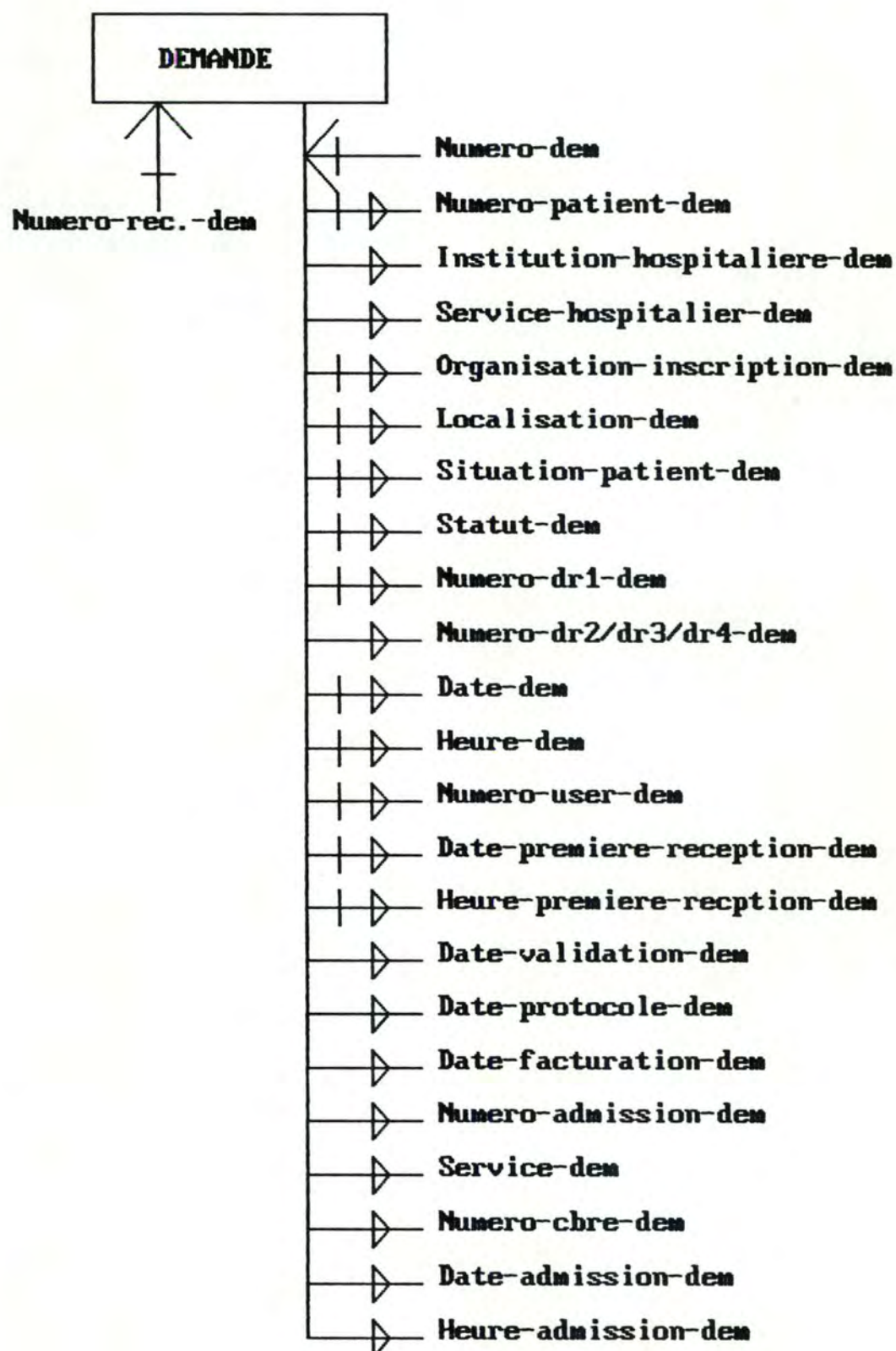
CLIN. St-JEAN	DOK/REF	REF	UPD. LEU.
LEFEBURE. P			

SCHEMAS DES ACCES NECESSAIRES

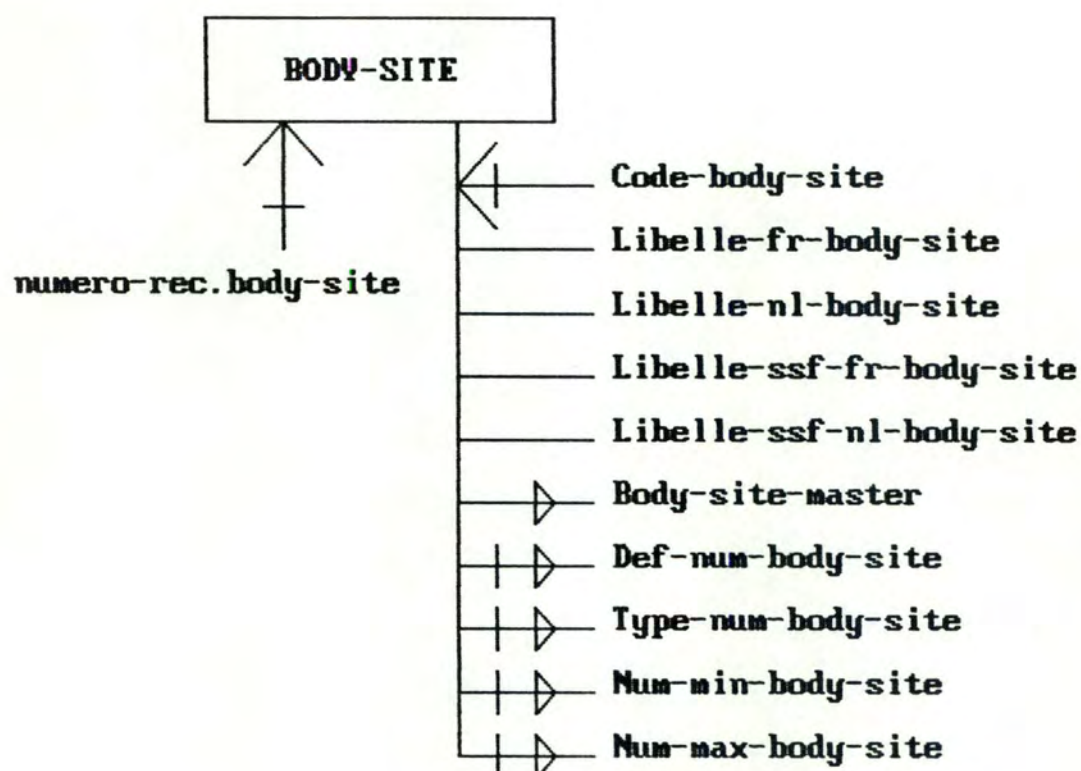
CLIN. St-JEAN	DOK/REF	REF	UPD. LEU.
LEFEBURE. P	EPIPATIE. DU; PC		



CLIN. St-JEAN	DOK/REF	REF	UPD. LEV.
LEFEBURE. P	EPI DEM. DU; PC		

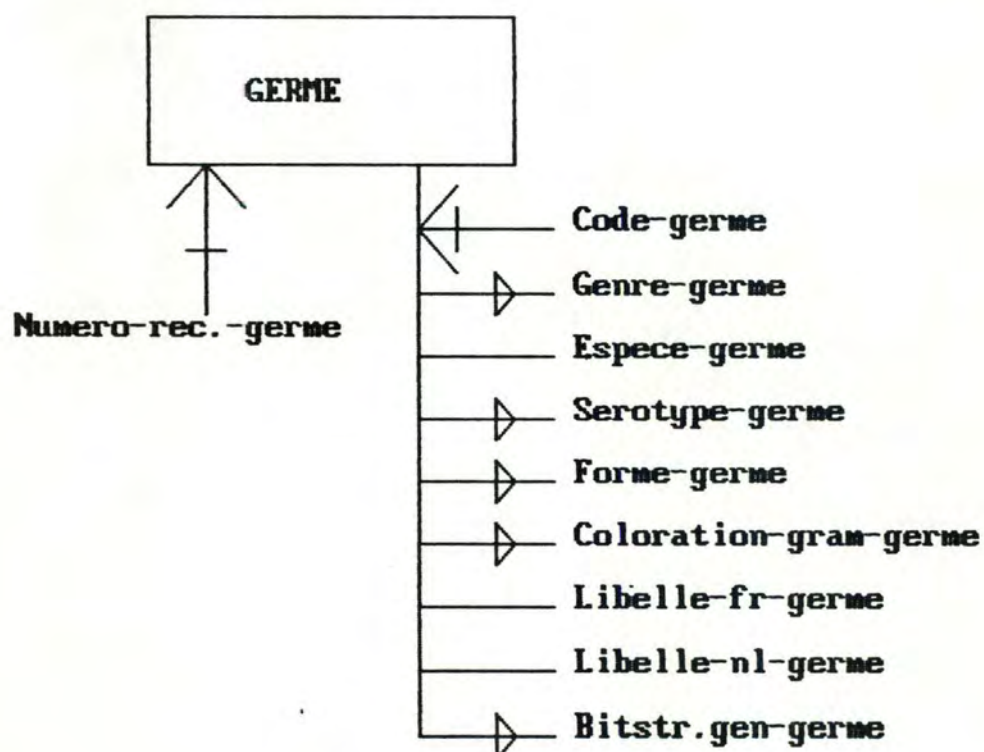
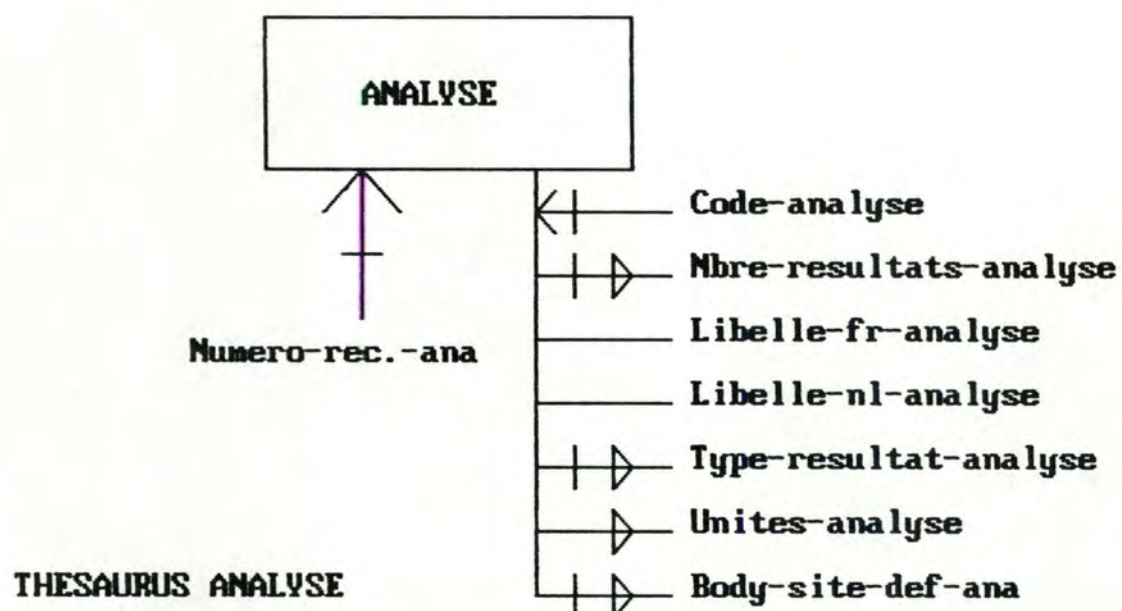


CLIN. St-JEAN	DOK/REF	REF	UPD. LEU.
LEFEBURE. P	EPI BOD. DU; PC		

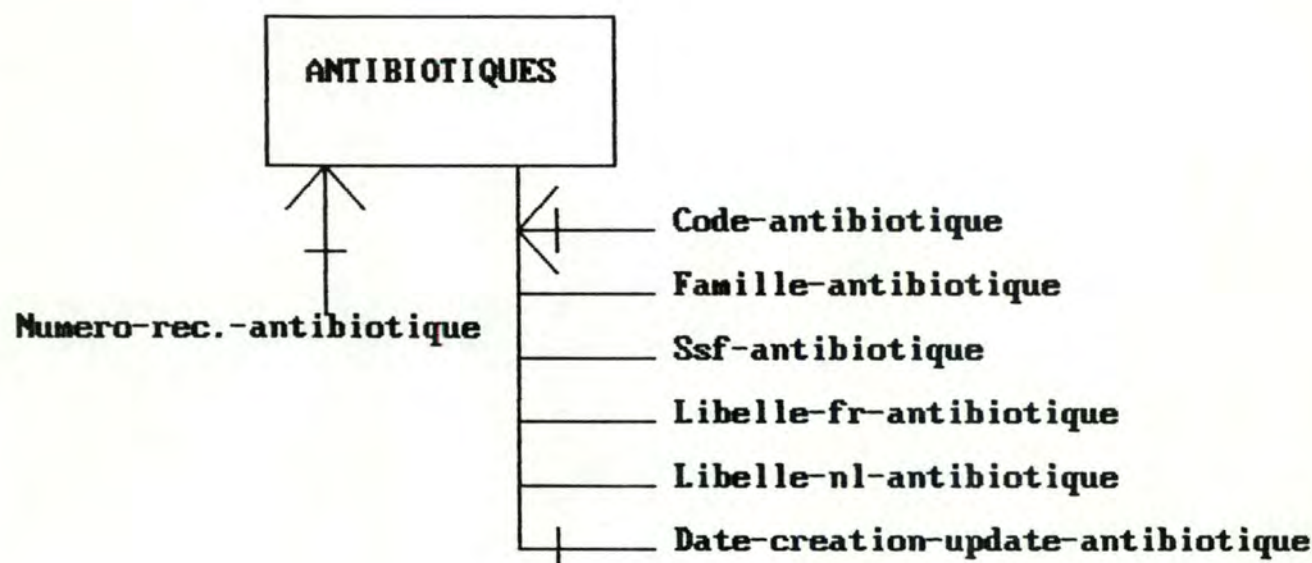


THESAURUS DE BODY-SITES

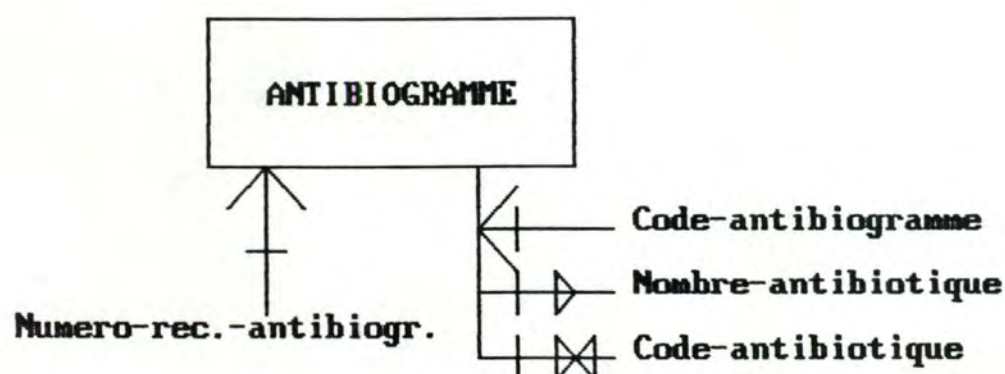
CLIN. St-JEAN	DOK/REF	REF	UPD. LEV.
LEFEBURE. P	EPIANA. DU; PC		



CLIN. St-JEAN	DOK/REF	REF	UPD. LEV.
LEFEBURE. P	EPI ANT. DU; PC		

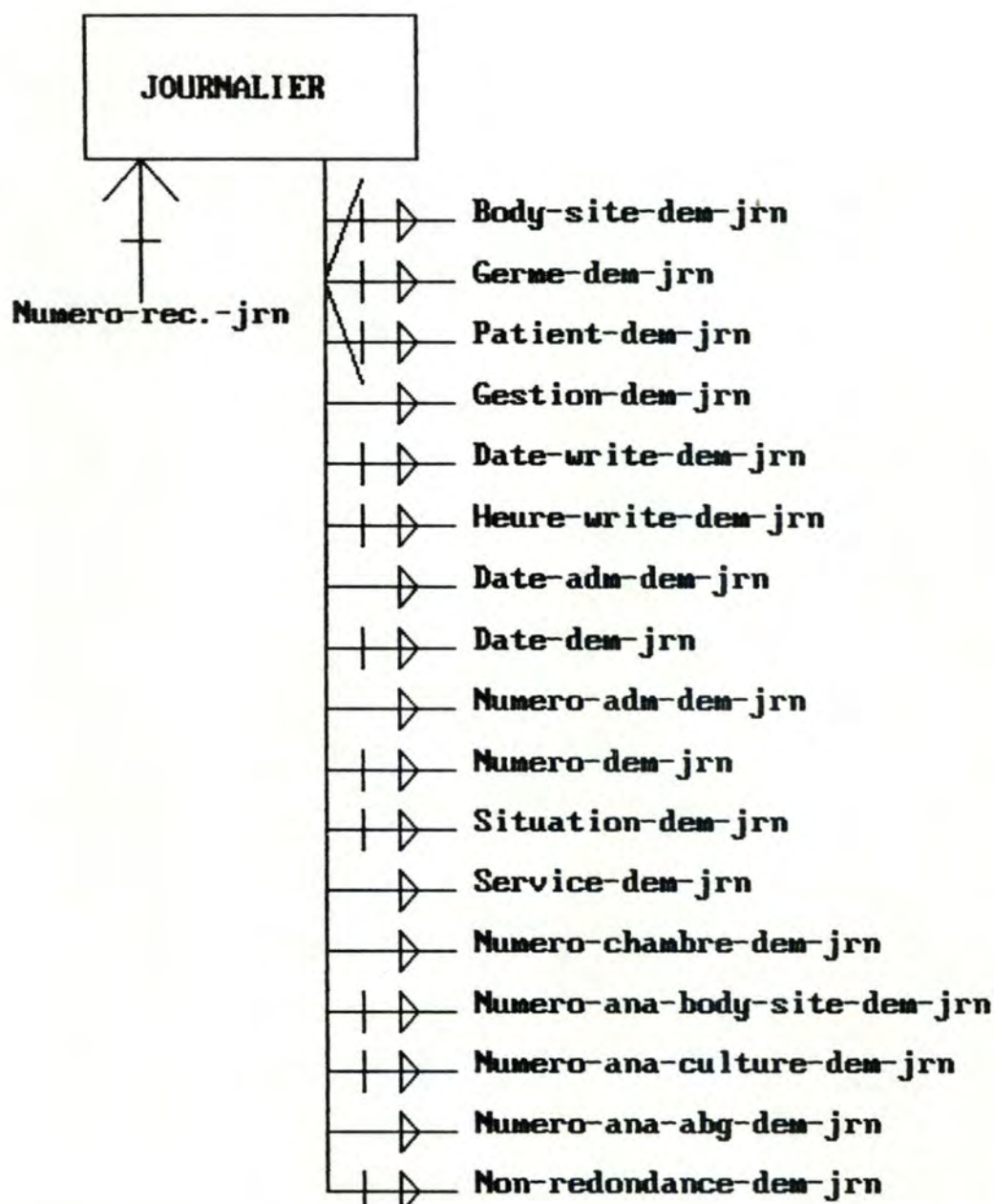


THESAURUS DES ANTIBIOTIQUES

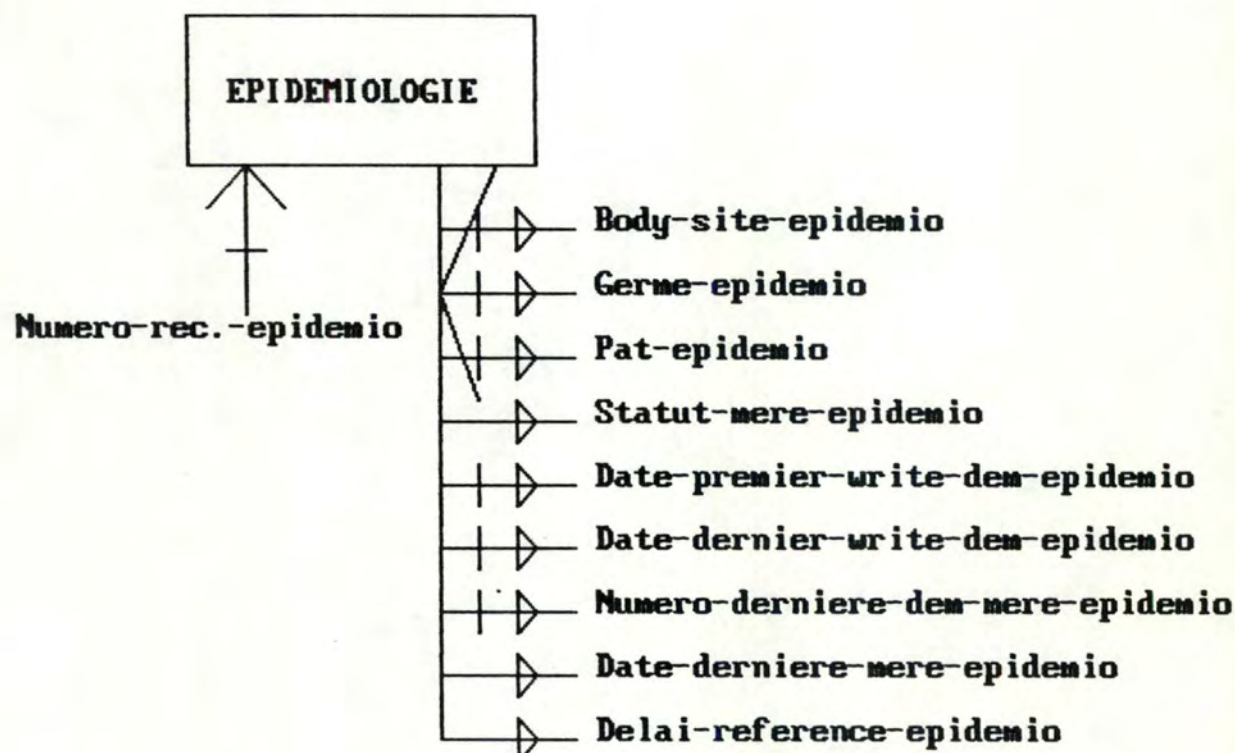


THESAURUS DES ANTIBIOGRAMMES

CLIN. St-JEAN	DOK/REF	REF	UPD. LEV.
LEFEBURE. P	EPI JRN. DU; PC		

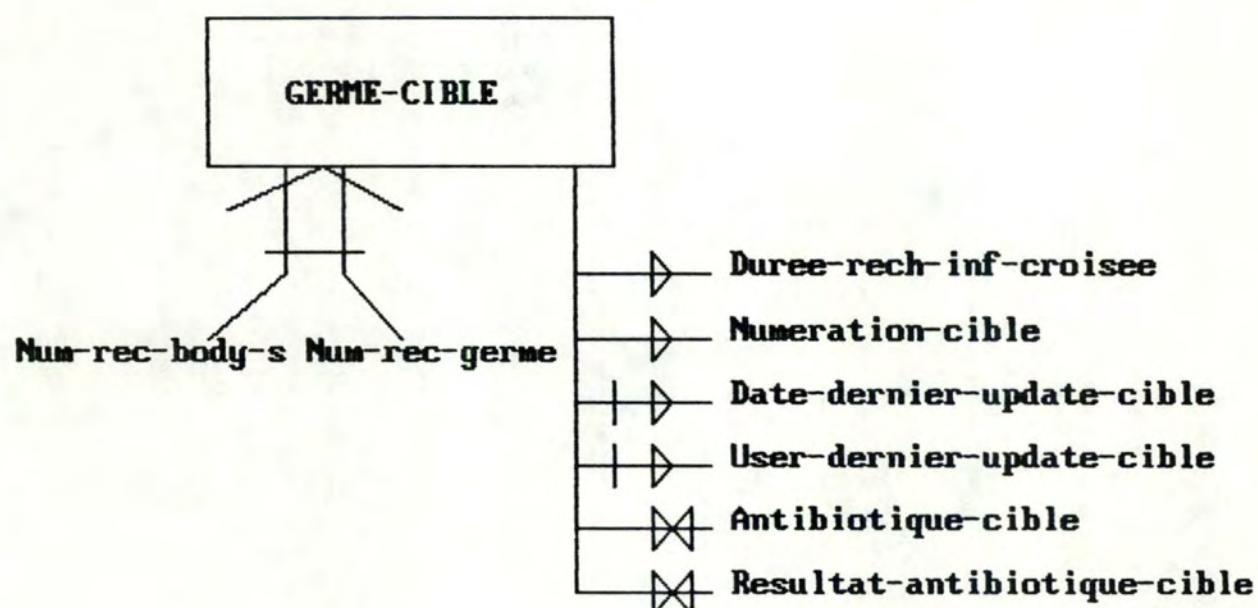


CLIN. St-JEAN	DOK/REF	REF	UPD. LEV.
LEFEBURE. P	EPI JRN. DU; PC		



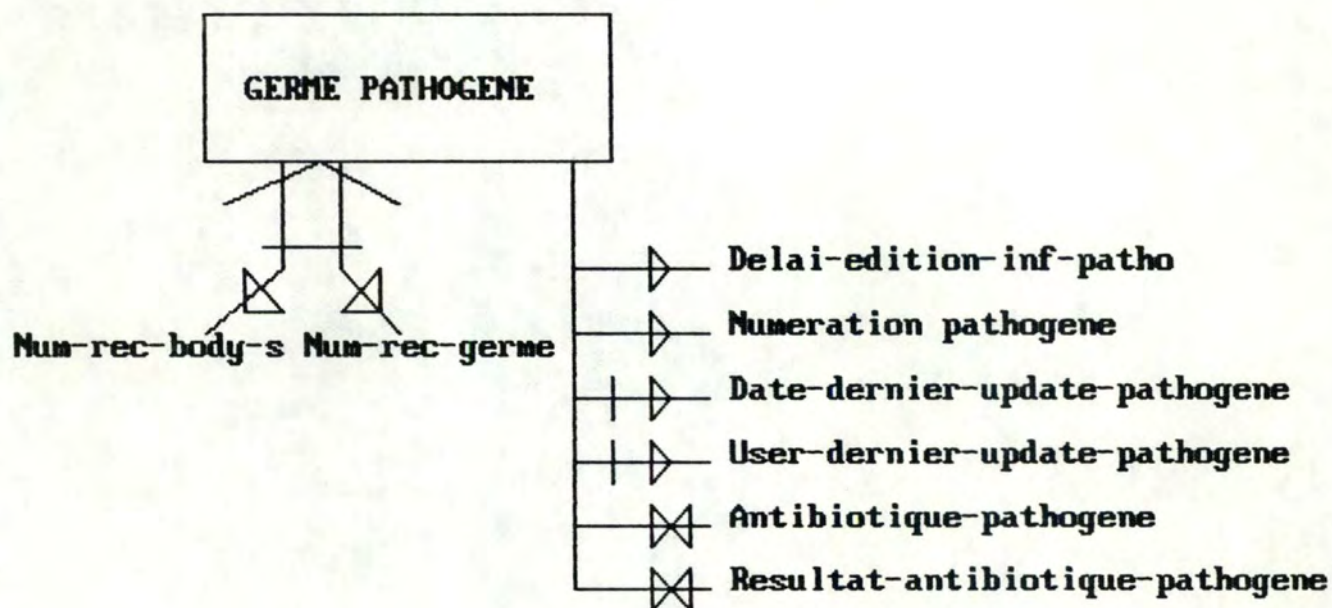
TESAURUS EPIDEMIO

CLIN. St-JEAN	DOK/REF	REF	UPD. LEU.
LEFEBURE. P	EPICIBLE. DU; PC		



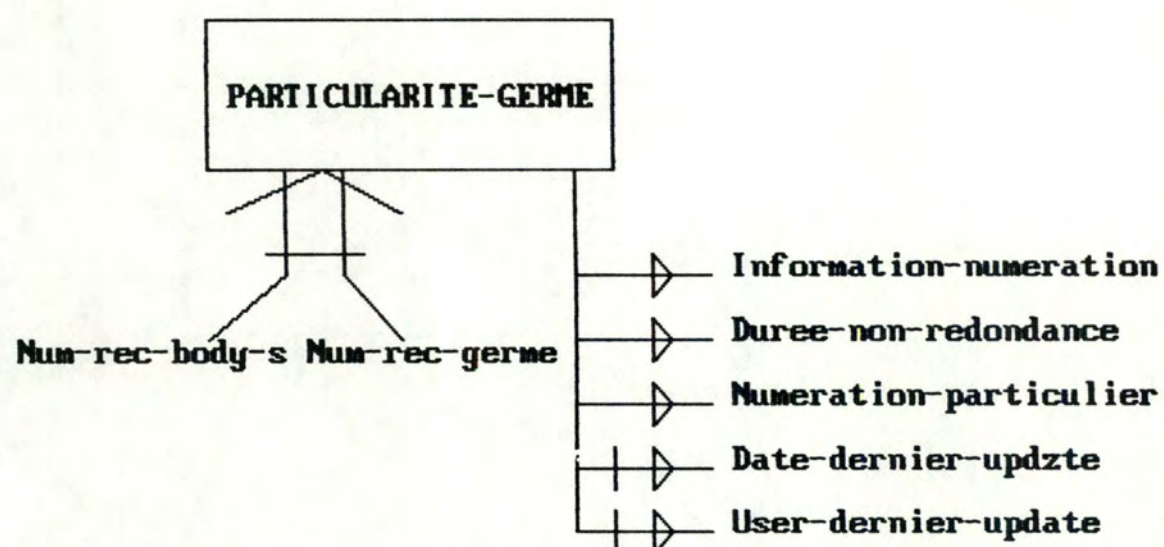
T.V.R des germes cibles derriere le THESAURUS des body-sites

CLIN. St-JEAN	DOK/REF	REF	UPD. LEV.
LEFEBURE. P	EPI PATH. DU; PC		



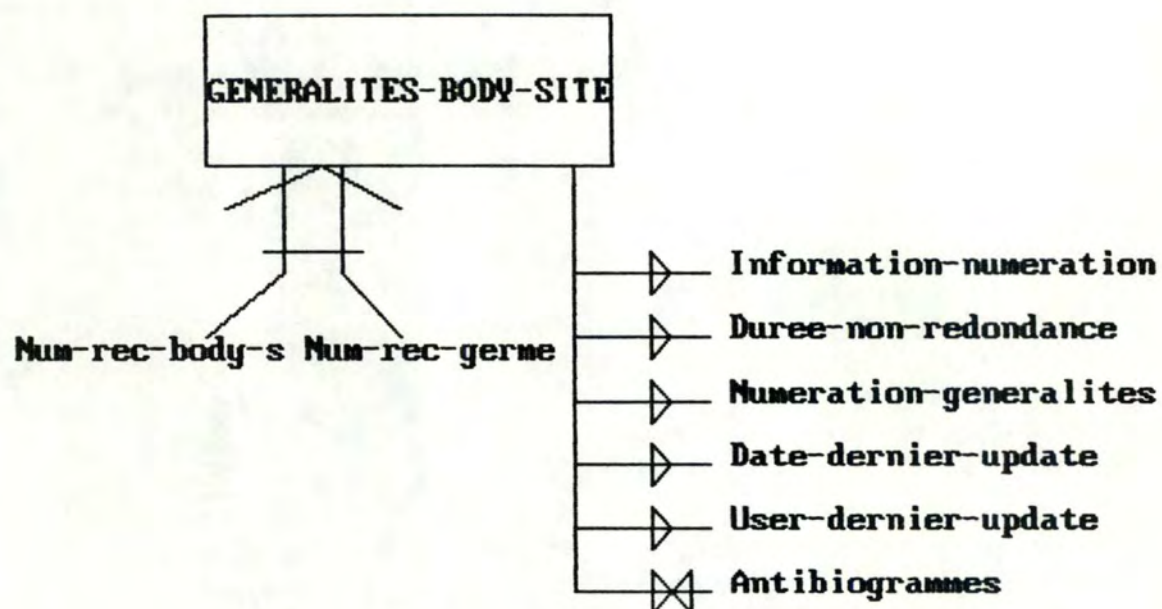
I.V.R des germes pathogenes derriere le THESAURUS des body-sites

CLIN. St-JEAN	DOK/REF	REF	UPD. LEU.
LEFEBURE. P	EPIPARTG. DU; PC		



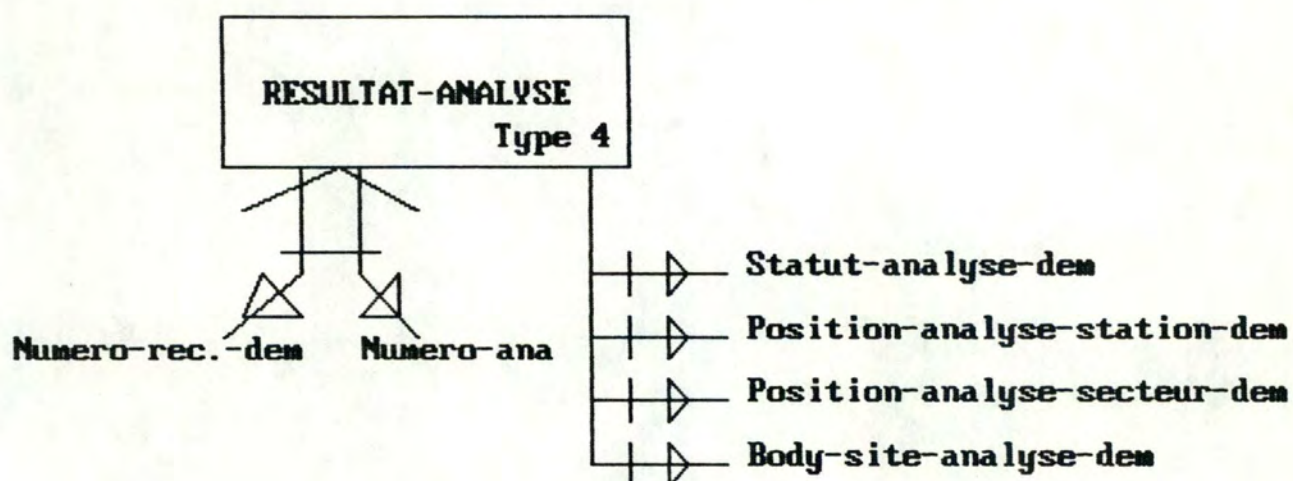
I.V.R des germes particuliers derriere le THESAURUS des body-sites

CLIN. St-JEAN	DOK/REF	REF	UPD. LEV.
LEFEBURE. P	EPI GEN. DU; PC		



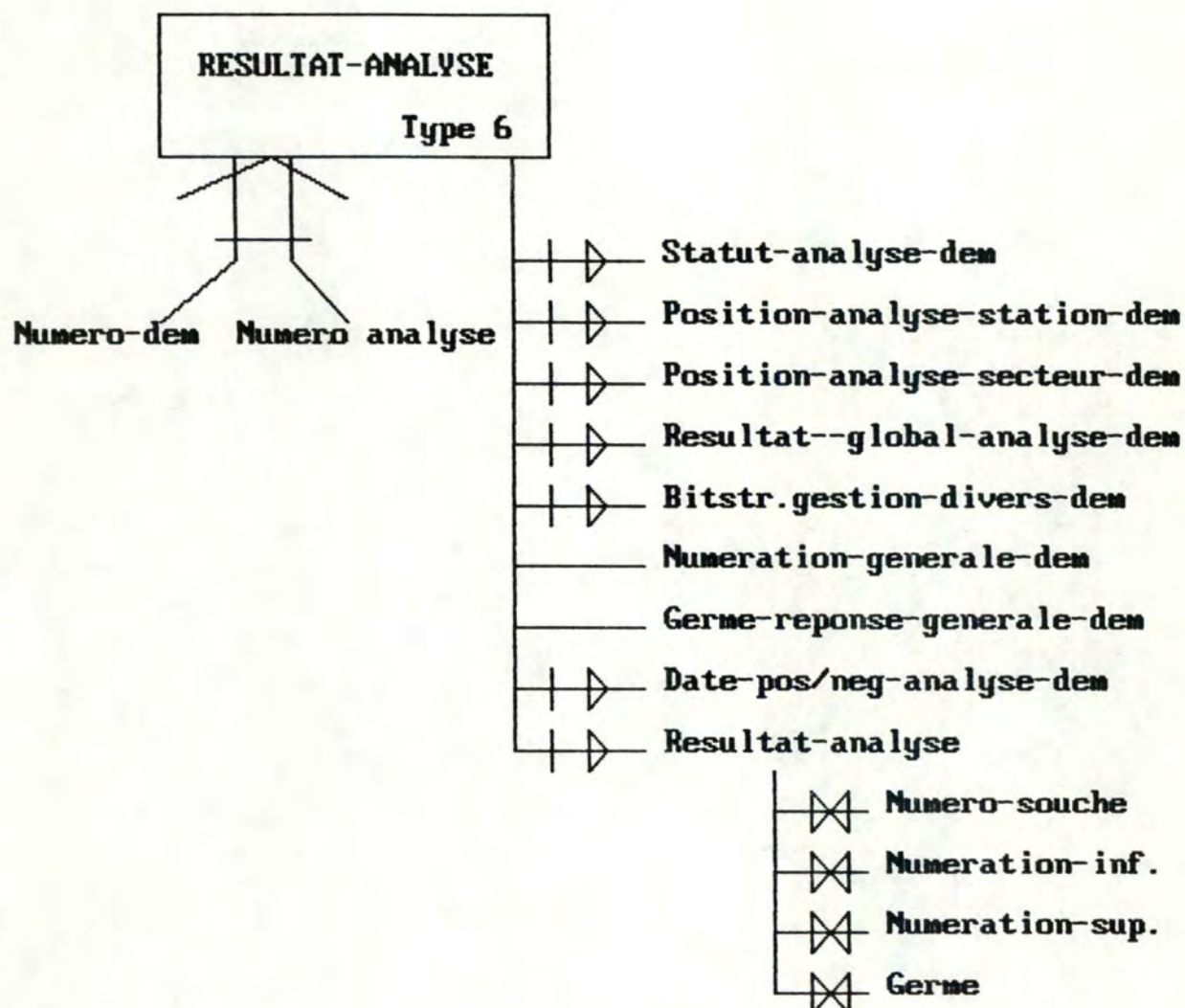
T.U.R des generalites derriere le THESAURUS des body-sites

CLIN. St-JEAN	DOK/REF	REF	UPD. LEU.
LEFEBURE. P	EPI RES4. DU; PC		



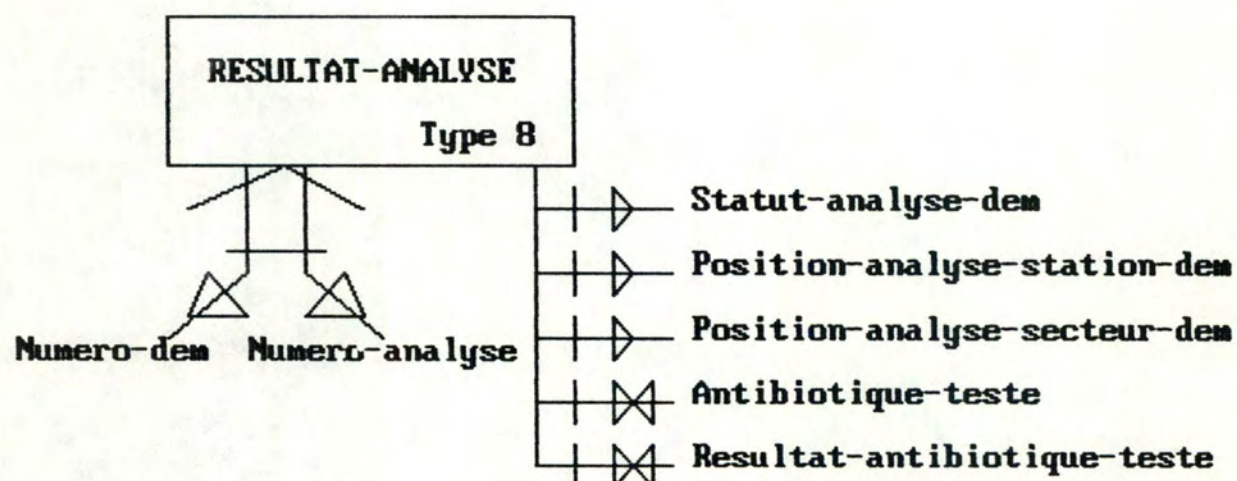
T.U.R des resultats analyse type 4 derriere le THESAURUS des demandes

CLIN. St-JEAN	DOK/REF	REF	UPD. LEU.
LEFEBURE. P	EPI RES6. DU; PC		



T.U.R des resultats analyse type 6 derriere le THESAURUS des body-sites

CLIN.St-JEAN	DOK/REF	REF	UPD.LEV.
LEFEBURE.P			



I.U.R des resultats analyse type 8 derriere le THESAURUS des body-sites

CLIN. St-JEAN	DOK/REF	REF	UPD. LEU.
LEFEBURE. P	EPI JRNEP. DU; PC		

